AN EVALUATION OF AN OMNI-
DIRECTIONAL (FISHEYE) VISION SENSOR FOR
ESTIMATING THE POSE OF AN OBJECT

A Thesis

Submitted to the Faculty of Graduate Studies and Research

In Partial Fulfillment of the Requirements

For the Degree of

Master of Applied Science

in

Software Systems Engineering

University of Regina

By

Shahin Parhizgar

Regina, Saskatchewan

March, 2014

Copyright 2015: S. Parhizgar
Shahin Parhizgar, candidate for the degree of Master of Applied Science in Software Systems Engineering, has presented a thesis titled, *An Evaluation of an Omni-Directional (Fisheye) Vision Sensor for Estimating the Pose of an Object*, in an oral examination held on December 8, 2014. The following committee members have found the thesis acceptable in form and content, and that the candidate demonstrated satisfactory knowledge of the subject material.

External Examiner: Dr. Xue Dong Yang, Department of Computer Science

Supervisor: Dr. Raman Paranjape, Software Systems Engineering

Committee Member: *Dr. Mehran Mehrandezh, Industrial Systems Engineering

Committee Member: Dr. Craig Gelowitz, Software Systems Engineering

Committee Member: **Dr. Luigi Benedicenti, Software Systems Engineering

Chair of Defense: Dr. Kelvin Ng, Environmental Systems Engineering

*via Video Conference
**Not present at defense
ABSTRACT

Two case studies of enhanced field of view are presented in this work. In the first an omni-directional lens is mounted on a single digital camera. In the second three perspective digital cameras are combined into a unique assembly (placed in row with known position to each another) to increase the sensing capabilities of the camera system.

To generate precise pose information from a fisheye camera, the unified/generic model is employed. The model is utilized to define the geometry of the camera. Omni-directional images captured are remapped to construct perspective images. A callback function for a mouse click event is developed to extract four non-coplanar feature points from remapped images of the target object. The image coordinates of the feature points are passed to a Pose from Orthography and Scaling with Iteration (POSIT) algorithm, to measure the objects pose in relation to the omni-directional system. The measured pose is exploited to calculate displacement and distance of the object from the camera. In this method, prior knowledge of the objects geometry is required.

To calculate 3-d position from corresponding images, the 3-camera imaging system exploits Epipolar geometry and the Longuet-Higgins algorithm. In this technique in addition to individual camera calibration, position and orientation of each camera with respect to other cameras in the system are necessary.
The experiments in this thesis are designed to measure the practicality of employing omni-directional lenses to generate metric information. In these experiments, measurements conducted via the 3-camera imaging system (control values) are compared against measurements gathered using the omni-directional imaging system.
ACKNOWLEDGEMENT

I sincerely acknowledge Dr. Raman Paranjape, the supervisor, for his guidance, his practical ideas, and his support. I also acknowledge Dr. Mehran Mehrandezh, for his insightful and out of the box view of engineering problems. I would also like to acknowledge the University of Regina, the Faculty of Engineering, and the Faculty of Graduate Studies and Research for their support.
DEDICATION

My deepest gratitude goes to my parents, sister, aunt, uncle, and dearest Arbely for their unconditional love, advice, and support throughout my journey away from home.
TABLE OF CONTENTS

ABSTRACT ..................................................................................................................... I

ACKNOWLEDGEMENT ................................................................................................. III

DEDICATION ................................................................................................................ IV

TABLE OF CONTENTS ................................................................................................. V

LIST OF TABLES ........................................................................................................ VIII

LIST OF FIGURES ........................................................................................................ IX

NOMENCLATURE ......................................................................................................... XI

ACRONYMS .................................................................................................................. XIV

1. INTRODUCTION ...................................................................................................... 1

1.1 Thesis organization ............................................................................................... 4

2. BACKGROUND ......................................................................................................... 6

2.1 Omni-directional cameras ..................................................................................... 6

2.1.1 Multi-camera or poly-dioptic imaging systems .............................................. 7

2.1.2 Rotating imaging systems ................................................................................ 8

2.1.3 Catadioptric cameras ....................................................................................... 8

2.1.4 Wide-angle/ Fisheye cameras ......................................................................... 9
2.2 From pinhole to omni-directional cameras ........................................ 12

2.2.1 The relation of the lens and the imaging plane ..................... 14

2.2.2 The principal point and the center of the imaging plane .... 14

2.2.3 Distortion .............................................................................. 15

2.2.4 Radial distortion and symmetry ........................................ 17

2.2.5 Single center of projection ................................................... 19

2.2.6 Square picture elements ..................................................... 21

2.3 Camera models ....................................................................... 23

2.3.1 Pinhole model + Distortion model ........................................ 23

2.3.2 Geometric projection models ............................................. 26

2.3.3 Fisheye projection model + Distortion parameters ............. 27

2.3.4 Alternative models .............................................................. 28

2.3.5 Unified/ Generic models ...................................................... 30

2.4 Camera calibration ................................................................... 34

2.5 Metric information from 2-d images ....................................... 40

2.5.1 Measuring pose ...................................................................... 40

3. METHODOLOGY ............................................................................ 42

3.1 Fisheye lens camera calibration ............................................... 46

3.2 Multiple-camera system calibration .......................................... 50
3.3 POSIT algorithm ........................................................................................................... 52
3.4 Position from corresponding images ........................................................................ 58

4. EXPERIMENTS AND ANALYSIS .................................................................................60
4.1 Experiment 1: Field of view ...................................................................................... 60
   4.1.1 Configuration 1 .................................................................................................. 67
   4.1.2 Configuration 2 .................................................................................................. 70
   4.1.3 Configuration 3 .................................................................................................. 72
4.2 Experiment 2: The POSIT algorithm ......................................................................... 74
4.3 Experiment 3: Fisheye camera and the POSIT algorithm ........................................ 80
4.4 Experiment 4: Analyze the accuracy of the fisheye camera ..................................... 83
4.5 Experiment 5: The 3-camera imaging system ......................................................... 93
   4.5.1 The POSIT approach ....................................................................................... 97
   4.5.2 The Longuet-Higgins approach .................................................................... 99

5. DISCUSSION ..................................................................................................................103

6. CONCLUSION ...............................................................................................................105

7. REFERENCES ...............................................................................................................107

8. APPENDICES ...............................................................................................................114
LIST OF TABLES

Table 4.1. High quality camera................................................................. 67
Table 4.2. High quality fisheye lens.......................................................... 67
Table 4.3. The approximate fisheye Field Of View .................................... 68
Table 4.4. The approximate perspective FOV............................................ 69
Table 4.5. Wide-angle lens ........................................................................ 70
Table 4.6. The approximate wide angle FOV ............................................. 71
Table 4.7. Cheap off the shelf webcam ...................................................... 72
Table 4.8. Unknown/ undefined fisheye lens .............................................. 72
Table 4.9. The approximate cheap configuration FOV ............................... 73
Table 4.10. Perspective-view and POSIT algorithm ..................................... 78
Table 4.11. Fisheye camera and POSIT algorithm ...................................... 81
Table 4.12. Measurements using a perspective camera ............................... 85
Table 4.13. Measurements using a fisheye camera ..................................... 86
Table 4.14. Comparing the average error for both perspective and fisheye camera ..... 92
Table 4.15. Distance of the cube measured to camera 1 ............................. 98
Table 4.16. Stereo vision 1 and 2 ............................................................... 100
Table 4.17. Stereo vision 1 and 3 ............................................................... 101
Table 4.18. Comparing results ................................................................ 102
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1.1.</td>
<td>Enhanced field of view of the fisheye lens camera</td>
<td>3</td>
</tr>
<tr>
<td>Figure 2.1.</td>
<td>Omni-directional cameras</td>
<td>7</td>
</tr>
<tr>
<td>Figure 2.2.</td>
<td>Omni-directional cameras</td>
<td>9</td>
</tr>
<tr>
<td>Figure 2.3.</td>
<td>The principal point and the optical axis</td>
<td>15</td>
</tr>
<tr>
<td>Figure 2.4.</td>
<td>Barrel and pincushion distortion</td>
<td>18</td>
</tr>
<tr>
<td>Figure 2.5.</td>
<td>The unified imaging model</td>
<td>31</td>
</tr>
<tr>
<td>Figure 2.6.</td>
<td>Sampled images of the calibration plane</td>
<td>36</td>
</tr>
<tr>
<td>Figure 3.1.</td>
<td>Distortion correction</td>
<td>47</td>
</tr>
<tr>
<td>Figure 3.2.</td>
<td>B. Li 2-d calibration pattern</td>
<td>50</td>
</tr>
<tr>
<td>Figure 3.3.</td>
<td>Schematic diagram of the POSIT algorithm</td>
<td>53</td>
</tr>
<tr>
<td>Figure 3.4.</td>
<td>The four non-coplanar points</td>
<td>55</td>
</tr>
<tr>
<td>Figure 3.5.</td>
<td>The four non-coplanar Image points</td>
<td>55</td>
</tr>
<tr>
<td>Figure 3.6.</td>
<td>Proposed approach to enhance the camera’s field of view</td>
<td>57</td>
</tr>
<tr>
<td>Figure 4.1.</td>
<td>The image sensor and circular image</td>
<td>63</td>
</tr>
<tr>
<td>Figure 4.2.</td>
<td>From full-circular image to a perspective image</td>
<td>63</td>
</tr>
<tr>
<td>Figure 4.3.</td>
<td>From full-circular image to desired output image</td>
<td>65</td>
</tr>
<tr>
<td>Figure 4.4.</td>
<td>Calculate perspective distance between cubes</td>
<td>75</td>
</tr>
<tr>
<td>Figure 4.5.</td>
<td>Enhance field of view via fisheye camera</td>
<td>80</td>
</tr>
<tr>
<td>Figure 4.6.</td>
<td>Corner extraction</td>
<td>84</td>
</tr>
<tr>
<td>Figure 4.7.</td>
<td>Failure to converge</td>
<td>88</td>
</tr>
</tbody>
</table>
Figure 4.8. Angle vs. error for the perspective camera ............................................. 89
Figure 4.9. Angle vs. error for the fisheye camera .......................................................... 89
Figure 4.10. Depth vs. error for the fisheye camera ....................................................... 91
Figure 4.11. The 3-camera imaging system being calibrated ........................................... 93
Figure 4.12. The view of the 3-camera imaging system ................................................... 94
Figure 4.13. The panoramic .............................................................................................. 94
Figure 4.14. The 5 IR reflectors rail ................................................................................ 99
Figure 8.1. Different image sensor formats ..................................................................... 119
NOMENCLATURE

\( pp \) : Principal point

\((x_{pp}, y_{pp})\) : coordinate of principal point

\( C \) : Camera center (center of the camera and used to define the camera coordinate frame)

\( X \) : Point in 3-dimensional space

\( x \) : Point in 2-Dimensional space (on the imaging plane)

\( f \) : Focal length of the camera

\( f_x \) : Focal length along the x axis

\( f_y \) : Focal length along the y axis

\( R \) : Rotation matrix

\[
\begin{pmatrix}
R_{11} & R_{12} & R_{13} \\
R_{21} & R_{22} & R_{23} \\
R_{31} & R_{32} & R_{33}
\end{pmatrix}
\]

\( t \) : Translation vector

\[
\begin{pmatrix}
t_x \\
t_y \\
t_z
\end{pmatrix}
\]
\( P \) : Projection matrix

\[
\begin{pmatrix}
R_{11} & R_{12} & R_{13} & t_x \\
R_{21} & R_{22} & R_{23} & t_y \\
R_{31} & R_{32} & R_{33} & t_z
\end{pmatrix}
\]

\( K \) : Calibration/ Camera Matrix (assume square pixels)

\[
\begin{bmatrix}
f_x & 0 & u_0 \\
0 & f_y & v_0 \\
0 & 0 & 1
\end{bmatrix}
\]

\((u_0, v_0)\) : Coordinates of the principal point in pixels

\((x_d, y_d)\) : Original distorted location/ coordinate

\((x_u, y_u)\) : Un-distorted location (corrected image coordinate)

\((x_{pp}, y_{pp})\) : Center of distortion (coordinate)

\(k_i\) : Ith radial distortion coefficients

\(\rho_i\) : Ith de-centering distortion coefficients

\(\alpha\) : Angle between the optical axis (principal axis) and the incoming ray (the incident angle measured in radians)

\(\beta\) : Reflection angle

\(W\) : Word frame

\(u\) : Horizontal image coordinate

\(v\) : Vertical image coordinate
\[
\begin{pmatrix}
X_c \\
Y_c \\
Z_c
\end{pmatrix}
: \text{ Coordinates of a point measured in the scene coordinate system}
\]

\[
\begin{pmatrix}
X_w \\
Y_w \\
Z_w
\end{pmatrix}
: \text{ Coordinates measured in the camera coordinate system (projective space)}
\]

\[r\] : Distance \((u, v)\) from the center (un-distorted)

\[r_d\] : Radial distance (in distorted images)

\[r_u\] : Radial distance (in un-distorted images)

\[B^T_A\] : Homogeneous transformation matrix of frame \(A\) relative to frame \(B\)
## ACRONYMS

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>POSIT</td>
<td>Pose from Orthography and Scaling with Iteration</td>
</tr>
<tr>
<td>2-d</td>
<td>2 Dimensional</td>
</tr>
<tr>
<td>3-d</td>
<td>3 Dimensional</td>
</tr>
<tr>
<td>SVP</td>
<td>Single effective View Point</td>
</tr>
<tr>
<td>DOF</td>
<td>Degree(s) Of Freedom</td>
</tr>
<tr>
<td>6DOF</td>
<td>6 Degrees Of Freedom</td>
</tr>
<tr>
<td>COD</td>
<td>Center Of Distortion</td>
</tr>
<tr>
<td>PFET</td>
<td>Polynomial Fish-Eye Transform</td>
</tr>
<tr>
<td>CCD</td>
<td>Charge-Coupled Device</td>
</tr>
<tr>
<td>CMOS</td>
<td>Complementary Metal-Oxide Semiconductor</td>
</tr>
<tr>
<td>IR</td>
<td>Infra-Red</td>
</tr>
<tr>
<td>FOV</td>
<td>Field of View</td>
</tr>
<tr>
<td>USB</td>
<td>Universal Serial Bus</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>UAV</td>
<td>Unmanned Aerial vehicle</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

A conventional perspective camera senses the visible spectrum of light, and can be exploited to extract meaningful information from the images it capture; information such as corner extraction, edge detection, pattern recognition and much more. Calibrate the perspective camera and combine it with available algorithms, to measured different quantities such as distance of objects from one another, depth of an object from a camera, and construction of 3-d scenes.

With the advent of low-cost quality cameras, miniature PC boards, and advance vision software, growing number of industries rely on vision sensors to perform processes such as, inspection and analysis of an appliance, localization\(^1\) of a work piece relative to a known object in a work cell, and robot guidance. Imaging-based technologies in addition enable unmanned vehicles to see and avoid obstacles, detect movement, navigate accurately, and find objects. In the following examples, [1] and [2], autonomous systems utilize visual imagery obtained by a perspective digital camera to track and estimate the pose of objects.

However a key disadvantage of conventional perspective cameras is that it typically has a limited field of view (less than 60°). This fact restricts its effectiveness in a variety of vision applications. The field of view of a camera is enhanced as one shifts from a perspective camera to an omni-directional camera. Omni-directional cameras cover an extended field of view and are built using different methods. Because of this

\(^1\) Localization is used to determine one’s location (position) in an environment. After localization a robot or an autonomous vehicle can perform path planning and effective navigation.
they can have varied fields of view. In order to increase the field of view of an imaging system most efforts are centered on the use of stereo vision, rotating imaging systems, and cata-dioptic cameras.

The fisheye lens camera is another effective method to enhance the field of view. The combination of a camera plus a fisheye lens is likely lighter than any other omni-directional system. It also consumes less energy as oppose to rotating or multi-camera imaging systems. Light weight and low power consumption is an important factor when designing Unmanned Aerial Vehicles (UAV).

The catadioptric system has been built as such, that a blind spot is formed in front of the camera. The drawback of a rotating imaging system is caused by its moving parts and the need to be positioned accurately; it also requires adequate time to reconstruct an enhanced field of view. In multi-camera imaging systems, all cameras need to be synched, via hardware or software. There is also the need to have overlapping images from neighboring cameras to construct the enhanced field of view. It is evident using a fisheye lens to generate enhance field of view in certain circumstances is advantageous. In Figure 1.1 (Bottom), the camera captures an image using a fisheye lens. The same camera, not using the fisheye lens, is used in Figure 1.1 (Top) to image the cubes, as the distance of the objects to the camera has remained the same.
Figure 1.1. Enhanced field of view of the fisheye lens camera

The view from a conventional camera (Top) the view from a fisheye lens camera (Bottom). (Original in color)
1.1 Thesis organization

The following explains how this thesis has been organized and structured: to begin with, different types of omni-directional cameras have been introduced. Other topics discussed are the increased field of view of the fisheye lens camera and the undesirable side effects, the need for a suitable geometric model to define the fisheye lens camera, camera calibration, distortion correction, and extraction of metric information from images.

Afterwards in the methodology section, specifics which explain how the equipments are utilized to conduct the experiments, the calibration toolboxes employed to calibrate the fisheye lens camera and the multi-camera system, and the algorithms exploited to extract metric information from the 2-d images are given.

The first experiment attempts to quantify the limitations that are inherited in the fisheye lens imaging system. In the second experiment a perspective camera is used to run the POSIT algorithm and measure pose of known objects. In the third experiment after converting the omni-directional image into a perspective image, the POSIT algorithm is employed to calculate pose.

In the fourth experiment large sets of measurements are managed to analyze the accuracy of the POSIT algorithm, when using the fisheye lens camera. For the last experiment, experiment five; an enhanced field of view has been constructed by combining three perspective cameras into a unique assembly. The measurements collected via the multi-camera imaging system (control values) are compared against
the measurements gathered using the fisheye camera. Last but not least a conclusion, contributions and possible extensions are proposed and presented.
2. BACKGROUND

A conventional perspective camera:

- Can be easily bought at any electronic store.
- The pinhole model is a fair approximation for them.
- They obey perspective projection.
- For the most part have a uniform resolution.
- Have a small field of view (usually around 60°).

2.1 Omni-directional cameras

As mentioned in the introduction, a disadvantage of using a conventional perspective camera is its limited field of view; hence using an omni-directional camera is desirable. Wide-angle, fisheye, and cata-dioptic cameras are all synonymous with omni-directional imaging systems.

Enhanced field of view can also be constructed, employing more than one conventional perspective camera combined into a unique assembly or utilizing a rotating imaging system. Both imaging systems are not always referred to as omni-directional cameras in literature; nevertheless they do cover an extended field of view and are categorized as omni-directional systems.
2.1.1 Multi-camera or poly-dioptic imaging systems

A larger field of view can be constructed by combining a number of perspective cameras (Figure 2.1). In addition to having the intrinsic parameters of each camera, for this setup to function and perform 3-d measurements, it is also essential to have knowledge of the pose (rotation and translation) of each camera in relation to the other cameras in the system. The term poly-dioptic is used in [3] to define such cameras.

Figure 2.1. Omni-directional cameras

Spherical digital video camera system; image duplicated from [4] (poly-dioptic camera) each camera pointing in a different direction (Left) 3-sensor stereo cameras; image duplicated from [4] (Right). (Original in color)

In multi-camera systems the centers of projection reside inside their respective lenses [5], and so multiple centers of projection exist (one for each camera). Furthermore when combining perspective cameras; it is necessary to overlap the fields of view in order to perform Epipolar geometry.
2.1.2 Rotating imaging systems

Another approach to enhance the field of view is to rotate the entire imaging system, about its center of projection [5] (Figure 2.2 (Right)). The sequence of images acquired by the camera are “stitched” together to obtain a large field of view.

The disadvantage of having a rotating imaging system is that it requires moving parts and precise positioning. Another drawback lies in the total time required to obtain an image in this approach.

2.1.3 Catadioptric cameras

Explained in [3], the camera is introduced in 1990 by Yagi and Kawato [6]; catadioptric cameras combine a conventional camera with a shaped mirror — such as a parabolic, hyperbolic, or elliptical — and provide 360° field of view in the horizontal plane and more than 100° in elevation (Figure 2.2 (Left)). Although the central catadioptric cameras and fisheye cameras have different physical constructions, they are not too different from the viewpoint of mathematical modeling [7]. Therefore, it should not come as a surprise that many fisheye camera models are in fact the revised or enhanced versions of catadioptric camera models.
2.1.4 Wide-angle/ Fisheye cameras

On the other hand one can employ a wide-angle/ fisheye lens in place of a conventional camera lens to enhance the field of view. The wide-angle/ fisheye lens camera has a short focal length\(^2\) and its field of view is approximately \(180^\circ\) (the field of view can be slightly more than a hemisphere). The terms ‘dioptic omni-directional cameras’, in [3], and ‘pseudo panoramic cameras’, in [9], are also used to define such cameras.

---

\(^2\) The field of view of a camera is a function of its focal length. A wide-angle lens has a small focal length, a telephoto lens has a large focal length, and a zoom lens has an adjustable focal length [12].
The introduction of wide-angle/ fisheye lenses to enhance the field of view has adverse side effects. These side effects are briefly described here, as additional details are provided later in the thesis.

**Fisheye lenses tend to deviate from the Single effective View Point (SVP) or the single center of projection property.**

Constructing a fisheye lens that has, a single center of projection is difficult. However fisheye lenses are now easier to produce and can better approximate the single center of projection property.

**Fisheye lenses introduce considerable radial distortion.**

One of the biggest challenges while using a fisheye lens is dealing with its severe radial distortion, as it causes image points to shift, making straight lines appear curved.

**The captured image has an overall lower geometric resolution.**

Another drawback of fisheye lens cameras is that the camera has an overall lower spatial resolution. For example a 1/3” image sensor is employed to capture 60° in a conventional perspective camera. In a fisheye lens camera the same size image sensor is capturing 180°. This produces an inferior imaging quality in comparison to a perspective camera.
In the image, resolution decreases non-linearly towards the edge of the frame.

A fisheye lens produces variable resolution images. The image captured via the fisheye lens has higher resolution fovea, and resolution decreases non-linearly toward the peripheries of the image.

The simple pinhole geometry is no longer valid for cameras using a fisheye lens.

Fisheye lens imaging systems are designed to cover roughly 180° in front of the camera. Yet it is impossible to project this hemisphere on an imaging plane by perspective projection. Therefore a fisheye lens camera cannot be defined using the pinhole geometry and modeled via perspective projection, a relatively simple practice. A new camera model is required to define the geometry of the fisheye lens camera.

The projective models of fisheye lenses vary from camera to camera.

Unlike the three types of central catadioptric cameras, the projective models of fisheye lenses, vary from camera to camera and depend on the lens field of view [3]. As a result it cannot be represented via an exact parametric function and a unified imaging model is needed to define them.

---

3 When angle of incoming rays and the optical axis approach pi/2 (90°) the perspective model (stops working) projects points infinitely far and it is not possible to remove this singularity with the conventional distortion models [7], and [23].
2.2 From pinhole to omni-directional cameras

The pinhole camera model cannot be applied to define the geometry of a fisheye lens camera. However the pinhole camera model is a practical framework to approach, understand, and deal with challenges that concern enhanced field of view.

In addition the pinhole camera model is a great way to introduce the essentials of image acquisition and camera calibration. The basics needed to interpret physical scenes and to exploit the camera as a 3-d measuring device is also conveniently explained with the pinhole camera model.

However the pinhole camera, as it is, requires a very small opening to produce a sharp image. This in turn reduces the amount of light passing through the opening and thus leads to extremely long exposure times [10].

A glass lens or system of optical lenses can focus a large amount of light (ray) on a point to give one fast imaging [11]. The need for a glass lens or system of optical lenses is clear; however adding one can alter the geometry of the pinhole camera model. For this reason the “thin” lens model is introduced [10]. The “thin” lens model adds a lens to the pinhole model but keeps the geometry intact.
Now that the pinhole camera has a lens, the ideal camera is introduced. The ideal camera has the following properties:

- The lens is exactly parallel to the imaging plane.
- The principal point\(^4\) is equivalent to the center of the imaging plane.
- The lens has no distortion (mathematically ideal lens).
- The lens is radially symmetric.
- The camera has a single center of projection or a Single effective View Point (SVP property).
- The geometry of images formed on the imaging plane are the same for both horizontal and vertical lines (picture elements on the imaging plane are square).

Even though these assumptions are usually never true in practice, they provide researchers with a structure, to deal with different challenges concerning enhanced field of view. For example a calibration tool for an omni-directional camera can accept a false assumption mentioned above and deem its influence negligible. This practice is not unusual and it is performed in order to simplify the calibration process and emphasize on more influential factors (undesirable side effects).

---

\(^4\) The principal point or the optical center is where the optical axis intersects with the imaging plane. Optical center is the origin of the camera frame. Optical axis is the z-axis of the camera frame pointing outward.
2.2.1 The relation of the lens and the imaging plane

Taking a conventional perspective camera apart, the two main components in creating an image are:

- Image sensor/ sensor plane/ image plane/ imager/ or projective plane.
- Glass lens/ or system of optical lenses.

To understand the image sensor, think of it as a 2-dimensional plane composed of picture elements (also referred to as pixels). Each element is sensitive to light and will register lights intensity. The lens as mentioned before is required for fast imaging. In the ideal camera, the lens is exactly parallel to the imaging plane. In reality however it is not easy to mechanically align the lens (this includes a fisheye lens) and the imager. The result of this design imperfection, caused by the assembly process of the camera as a whole, is image distortion.

2.2.2 The principal point and the center of the imaging plane

In the ideal camera the principal point (where the principal axis intersects with the imaging plane, as shown in Figure 2.3) and the center of the imaging plane are equivalent. Whether using a conventional perspective or a fisheye lens, this assumption is not true since that means someone attached the optical axis of the camera to micron accuracy to the center of the imaging plane (as explained in [11]). The result of this misalignment is an offset that needs to be considered. Accounting for this displacement is particularly important when attempting to generate a perspective image, from an
omni-directional image, or running the POSIT algorithm (after generating the perspective image).

Figure 2.3. The principal point and the optical axis

2.2.3 Distortion

There are different forms of distortion that occur in imaging systems. Some have little effect and are overlooked (not even included in distortion classifications); others such as radial distortion need to be corrected. Typically distortion is divided into two main classification groups: radial and tangential distortions [12], and [10]. Others [13] have classified distortion into radial, de-centering, and thin prism distortion.

In [14] and in [12] de-centering distortion is synonymous with tangential distortion (most likely because it is the only form of tangential distortion they account for). However Strum et al. in [15] states that de-centering distortion is only a type of tangential distortion. In [16] de-centering distortion is described as misalignment of the individual lens elements, and the non-perpendicularity of the lens assembly and the image plane.
Hughes et al. [14] group the reason for tangential distortion as: Inaccurate distortion center estimation and thin prism distortion, and describes thin prism distortion as imperfections in lens design, manufacturing, and camera assembly, which produces both radial and tangential distortion. Mei et al. [17] reaffirms that improper lens and camera assembly generates both radial and tangential distortions.

It is clear by the leading paragraphs, that the classification of distortion especially tangential distortion is ambiguous. This ambiguity is produced because what creates a type of distortion can induce a bit of some other distortion; at the same time correcting one distortion can cause another distortion to be less significant.

The researcher in [16] deals with medium to wide-angle camera lenses and demonstrates that vast majority of tangential distortion can be compensated for by only using distortion center estimation; a viewpoint shared in [14]. This approach is valid for perspective camera lenses as Strum et al. [15] presents a camera model that neglects tangential distortion and supposes that image coordinates are centered in the distortion center and that the latter is equal to the principal point. Tsai [18] also introduced a camera model where he only accounts for radial distortion.

Radial distortion on the other hand is much better defined in literature and is explained in detail. Suffice to say, a fisheye lens produces severe radial distortion, which needs to be accounted for if the fisheye camera is to be utilized to measure pose of objects in the scene.
One last topic that needs to be discussed is the radial center of distortion. In several camera models it is assumed that the distortion center coincides with the principal point. Other models exist that consider them as distinct and/or include additional distortion terms [15].

2.2.4 Radial distortion and symmetry

A lens, as mentioned, is required in image acquisition. However by using a lens, radial distortion is generated as light rays pass through them before striking the imager. Radial distortion is particularly severe in fisheye lenses and needs to be corrected.

Displayed in Figure 2.3, in radial distortion, if image points are pulled closer along a radial axis from the center of distortion, the lens produces barrel distortion and if image points are pushed away from the center (typical of telephoto lenses) the lens produces pincushion distortion. Distortion at the optical center is considered minimum (zero) and increases as we move toward the peripheries of the image [11].

In Figure 2.3 the red rectangle is the undistorted image. The small blue rectangle represents barrel distortion. Pincushion distortion is the bigger blue rectangle. In a fisheye lens, distortion causes image points on the image plane to be displaced (pulled closer) in a nonlinear fashion from their ideal position in the pinhole camera model; along a radial axis from the center of distortion in the image plane [19].
Another variable that is considered when modeling lens distortion is radial symmetry. Consider the polar coordinate system (as opposed to the Euclidean coordinate system) to model radial distortion. If the pole represents center of distortion, the amount of distortion in a symmetrical lens depends on radial distance, and a polar angle has no effect.
A radially symmetric distortion can easily be modeled mathematically; however constructing such a lens is difficult\textsuperscript{5}. The consequence is a radial asymmetric distortion, where distortion varies not only with polar distance but also with polar angle. Most studies choose to disregard asymmetric distortion; However de Villiers [20] analyzes asymmetric distortion and accounts for it.

2.2.5 Single center of projection

A central camera or a camera with a Single effective View Point occurs when all light rays pass through a single optical center before reaching the image plane. In a non-central camera a locus of viewpoints in three dimensions, called a caustic, is formed [21].

A single center of projection is desirable, as captured images can be further processed by the large body of work in computational vision that assumes linear perspective projection. Keep in mind that the pinhole camera model is a central model. A single center of projection is also desirable in omni-directional cameras, as it allows generation of pure perspective images\textsuperscript{6} (converting an omni-directional image to a perspective image).

The construction of a fisheye lens that approximates the central camera is difficult. Strictly speaking even conventional perspective cameras are non-central, \textsuperscript{5}In addition to errors with lens manufacturing, not having square picture elements can also contribute to asymmetric radial distortion.  
\textsuperscript{6}Strum et al. in [15] explains how for central camera models the back-projection function delivers the direction of the camera ray. Non-central camera models do not possess a single optical center. In that case, the back-projection operation has to deliver not only the direction but also the position of a camera ray, e.g., some finite point on the ray.
however in reality the pinhole camera model is a good approximation for them and acceptable in almost all situations.

In 1997 Nayar [5] indicates that constructing a fisheye lens with single center of projection is complex. By 2000 D. Scaramuzza, in [3] and [22], points out that because of new manufacturing techniques and precision tools, the fisheye lens is easier to produce. He further reveals that since 2005 these lenses have been miniaturized to the size of 1-2 centimeters and therefore modern fisheye lenses approximate the single effective viewpoint property better. Based on his explanations D. Scaramuzza introduced his calibration tool for catadioptric and wide-angle/ fisheye lens cameras.

J. Kannala et al. in 2006 treats his fisheye lens camera as having a single center of projection. He calibrates his camera and generates a perspective image from an omni-directional image successfully [23]. Hartley et al. in 2007 assumes the central projection for his fisheye lens camera and agrees that deviation of light rays from an effective single point of projection is slight [24].

The fisheye lens camera employed in Hartley et al. body of work is small in size. In miniature fisheye lenses with very short focal length, the single effective view point property is a reasonable approximation [25]. Kannala et al. likewise argues that certain non-central cameras in practice are negligibly small compared to the viewed region so they are effectively “point-like” [23].
P. Cork in his book [12] explains that in theory, one cannot create a perspective image from a non-central wide-angle image but in practice if the caustic is small the parallax errors introduced into the perspective image will be negligible.

To conclude this matter, even though modern fisheye lenses are non-central, they approximate the single view point property and for small fisheye lenses (like the fisheye lens used in this work) the single view point property is even a better approximation\(^7\).

2.2.6 Square picture elements

Picture elements can be rectangular with non-orthogonal sides as oppose to square, which causes the geometry of images formed on the imaging plane to be different for horizontal and vertical lines. In addition, if the pixels are not square, then distortion is still radial, but the radial distortion function is not symmetric [24].

This irregularity needs to be accounted for in camera calibration; consequently making the math behind calibration a bit more complex (as shown in Appendix A - square picture elements with right angles simplifies the affine transform part of camera calibration). Remember before calibration, usually the aspect ratio\(^8\) of picture elements is unknown (see Appendix A for details).

Another challenge with rectangular picture elements is focal length estimation. Since the focal length cannot be measured directly (using a measuring device), after

\(^7\) The ORIFL190-3 (the fisheye lens employed in this research) lens body is small, at only 24 mm nominal diameter [50].
\(^8\) Aspect ratio is the width of a picture element to its height.
camera calibration, two focal lengths one in the horizontal and one in the vertical direction of the imaging plane are calculated (for a rectangular picture element); and the POSIT algorithm applied in this work only takes into account one focal length.

Also for certain image sensors (CCD chips), spacing between picture elements in the vertical and horizontal direction can be different, again causing an image to be unequally scaled in the horizontal and vertical directions. The effect of this distortion, as mentioned in [26], is for circles to appear as ellipses in the image.

Fortunately in modern image sensors, the assumption of square picture elements is almost true. The image sensor used in this work has a square picture element which is verified by the OpenCV calibration toolbox (the focal length in the horizontal and the vertical direction is equal).
2.3 Camera models

A camera model describes how a 3-d scene is transformed into a 2-d image [26]. Many different camera models are available but this work only concentrates and introduces few of the more important ones (since a comprehensive introduction to camera models is out of the scope of this thesis).

2.3.1 Pinhole model + Distortion model

The pinhole model can be defined via two sets of parameters:

- The extrinsic parameters.
- The intrinsic parameters.

Equation 2.1, represents the projection $a$ of 3-d point $X = [X, Y, Z, 1]^T$ in the scene onto a 2-d imaging plane $x = [u, v, 1]^T$, where $K[R|t]$ includes both extrinsic and intrinsic parameters.

$$x = K[R|t]X$$

(2.1)

$[R|t]$ is the extrinsic parameter and is a combination of $R$, the rotation matrix and $t$, the translation vector (defined up to a scale factor). $K$ is the intrinsic parameter in the equation above and is defined as Equation 2.2.

$$K = \begin{pmatrix} f_x & s & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

(2.2)
The intrinsic camera parameters (matrix) usually include the effective focal length (in horizontal and vertical directions \((f_x, f_y)\)), scale factor \(s\), and the image center (the principal point \((u_0, v_0)\) in image coordinates\(^9\)) [27].

In the calibration matrix \(s\) allows for a pixel skew factor, which according to [15] can model non-rectangular pixels or synchronization errors in the image read-out. For modern cameras the skew term is almost always zero [11], allowing one to derive Equation 2.3 (assume square pixels, i.e., \(s = 0\) [15])

\[
\begin{align*}
  f_x &= f_y = f
\end{align*}
\]

(2.3)

Thus far a mathematically ideal perspective camera is modeled. However in reality, cameras deviate from the ideal imaging model. Therefore the pinhole camera model is often enhanced by addition of various terms to account for distortion.

Brown’s “plumb-line method” [28] is a well-known approach, correcting the radial and the de-centering distortions. Brown’s approach is based on the idea (“truism”) that in linear perspective projections, straight lines in 3-d space always appear straight in 2-d projections, thus any apparent curvature in an image of straight lines is due to lens distortion [29].

---

\(^9\) The origin of the image coordinate system is in the upper left corner of the image array [27].
Brown’s distortion model is presented using polynomials accordingly:

\[
x_u = x_d + \bar{x}(k_1 r^2 + k_2 r^4 + k_3 r^6 + \ldots) + (\rho_1 (r^2 + 2\bar{x}^2) + 2\rho_2 \bar{x}\bar{y})(1 + \rho_3 r^2 + \ldots)
\]

\[
y_u = y_d + \bar{y}(k_1 r^2 + k_2 r^4 + k_3 r^6 + \ldots) + (2\rho_1 \bar{x}\bar{y} + \rho_2 (r^2 + 2\bar{y}^2))(1 + \rho_3 r^2 + \ldots)
\]

\[
\bar{x} = x_d - x_{pp}
\]

\[
\bar{y} = y_d - y_{pp}
\]

\[
r = \sqrt{(x_d - x_{pp})^2 + (y_d - y_{pp})^2}
\]

In Equation 2.4, \((x_{pp}, y_{pp})\) is the center of distortion, \((x_d, y_d)\) is the original distorted location/coordinate, \((x_u, y_u)\) is the new un-distorted (or corrected) location/coordinate, \(k_i\) the \(i\)th radial distortion coefficients, and \(\rho_i\) the \(i\)th de-centering distortion coefficients.

To un-distort a perspective image 2 or 3 radial terms are applied. Typically 1 or 2 terms are employed to account for de-centering distortion (note that de-centering distortion is neglected by some researchers (zero de-centering terms are used)).

Brown’s distortion model is a popular approach and numerous academics have either applied it or utilized a modified version of it to remove distortion. The OpenCV calibration software [11] makes use of the pinhole camera model accompanied with Brown’s model to account for distortion. Note that in fisheye lens cameras the high levels of distortion produced by the lens cannot be removed employing Brown’s distortion model.
2.3.2 Geometric projection models

The perspective projection of a pinhole camera can be represented according to Equation 2.5:

\[ r_u = f \tan(\alpha) \]
\[ \alpha = \beta \]  

(2.5)

In this equation \( \alpha \) is the incidence angle and is equal to \( \beta \) the reflection angle. The projected radial distance from the principal point (on the imager)\(^{10} \) is represented by \( r_u \). For wide-angle or fisheye lens cameras the incidence angle is not equal to the reflection angle \((\alpha \neq \beta)\). Taking into account the refraction caused by the lens, wide-angle/ fisheye lens cameras usually obey one of the following projections (with Equidistance projection being the most common):

**Equidistance projection (equiangular)**

\[ r_d = f(\alpha) \]  

(2.6)

**Stereographic projection**

\[ r_d = 2f \tan(\alpha/2) \]  

(2.7)

**Orthogonal projection (sine-law)**

\[ r_d = f \sin(\alpha) \]  

(2.8)

---

\(^{10} \) In the projection above straight lines in the scene are projected as straight lines on the image plane (Rectilinear). Note this projection only works for narrow-angle images. So unless objects lie near the optical axis the shape and intensity of objects are distorted [57].
Equisolid angle projection

\[ r_d = 2f \sin(\alpha / 2) \]  \hspace{1cm} (2.9)

The following equations assume projection to be rotationally symmetric (from the optical axis) and map the angle between the incoming light ray and the optical axis to some distance from the image center.

2.3.3 Fisheye projection model + Distortion parameters

In practice the projection curve, in wide-angle/ fisheye lenses, deviates from its intended geometric model. The work in [9], investigates the addition of the Brown-parameters to the basic geometric fisheye model to compensate for the remaining “systematic effects”. In [14], it is proposed that the distortion function can be appended with polynomial elements, Equation 2.10, to account for the deviations of the lens from the projection function (the polynomial in the equation is an odd order with its 0 order term removed):

\[ \Delta r_d = K_1 r_u^3 + K_2 r_u^5 + K_3 r_u^7 \]  \hspace{1cm} (2.10)

In Equation 2.10 the coefficients are represented by \( K_n \). To apply the polynomial, in order to better approximate real lenses, the geometric fisheye model is the first term in the series and \( \Delta r_d \) is then added. Accordingly the Equidistant projection function is\textsuperscript{11}:

\[ r_d = f \arctan \left( \frac{r_u}{f} \right) + \Delta r_d \]  \hspace{1cm} (2.11)

\textsuperscript{11} The distortion function in Equation 2.11 is derived as we solved for Equations 2.5 and 2.6 in terms of \( \alpha \).
2.3.4 Alternative models

Research on the wide-angle/fisheye lens camera models is extensive, and many alternative models exist. In this section, some of the more interesting models are briefly described:

**FishEye Transform (FET)**

The FishEye Transform is a logarithmic function proposed by Basu and Licardie [30]. In Equation 2.12, $s$ is a simple scalar and $\lambda$ controls the amount of distortion across the image.

$$r_d = s \log(1 + \lambda r_u)$$ (2.12)

**Field-Of-View model (FOV)**

Devernay and Faugeras in [31], propose the field-of-view model. They recommend complementing their model with the classical polynomial radial distortion model. In Equation 2.13, $\omega$ is the angular field-of-view of the ideal fisheye lens camera.

$$r_d = \frac{1}{\omega} \arctan \left( \frac{2 r_u \tan \omega}{2} \right)$$ (2.13)

**Odd-order polynomial (Standard polynomial model)**

Described by Slama [32], in Equation 2.14, $K_i$ is the $i$th polynomial coefficient.

$$r_d = r_u + \sum_{n=1}^{\infty} K_n r_u^{2n+1} = r_u + K_1 r_u^3 + K_2 r_u^5 + \cdots$$ (2.14)
**Division model**

Different versions, of the so-called division model, are available [33], and [34]. The division model as defined by Fitzgibbon is presented accordingly:

\[
r_u = \frac{r_d}{1 + \sum_{n=1}^{\infty} K_n r_d^{2n}} = \frac{r_d}{1 + K_1 r_d^2 + K_2 r_d^4 + \cdots}
\] (2.15)

**Polynomial FishEye Transform (PEFT)**

A polynomial-based model that uses both odd and even coefficients is proposed by Basu and Licardie [30] and separately by Fitzgibbon [33].

\[
r_d = \sum_{n=0}^{\infty} K_n r_u^n = K_0 + K_1 r_u^1 + K_2 r_u^2 + \cdots
\] (2.16)

Hughes et al. explain in [35] that the Odd-order polynomial and the Division models are insufficient to accurately describe the level of distortion introduced by wide-angle/ fisheye lenses. Consequently a polynomial that uses both odd and even coefficients can be utilized to accurately model the radial distortion introduced by a wide-angle/ fisheye lens. Basu et al. in Equation 2.16, allows 0\(^{th}\) order term and suggest in [30] that a fifth-order form of the model is adequate to simulate the radial displacement introduced by fisheye lenses.
2.3.5 Unified/ Generic models

Professional manufacturers of wide-angle/ fisheye lenses, for scientific/ industrial imaging applications, attempt to design lenses where the projection/ distortion curve approximates closely with one of the projection functions discussed above. Knowing what projection model to employ to define a (high-end) wide-angle/ fisheye lens is practical, however at times this information may not be available\textsuperscript{12}

There is also the desire to calibrate miniature, low-cost, fisheye lenses\textsuperscript{13}, with unknown/ undefined projection/ distortion curves. Additionally, while the three types of central catadioptric cameras (parabola, hyperbola, and ellipse) can be represented through an exact parametric function; the projective models of fisheye lenses vary from camera to camera depending on the lens field of view [3]. Therefore it is advantageous to have only one projection model suitable for different types of wide-angle/ fisheye lenses, and map a lens independent of its field of view.

**Spherical cameras and the unified model**

So far a 2-d projective plane is discussed. Although suitable for conventional perspective cameras, it is instead beneficial to consider a projective sphere for wide field of view imaging systems. As explained, in [12], the spherical camera is nothing more than a theoretical construct; however in the unified imaging model this powerful framework is exploited as an intermediate step to convert omni-directional images to

\textsuperscript{12} Attempt to calibrate a lens with a priori unknown projection function.
\textsuperscript{13} Bakstein and Pajdla introduced a spherical retina (and a radially symmetrical mapping between the incoming light rays and pixels in the image) with the goal to model off-the-shelf cheap lenses [26].
perspective images. In the unified imaging model, images captured via perspective, wide-angle/ fisheye, and catadioptric cameras can be mapped onto a sphere where one now can treat them in a uniform manner.

Figure 2.5. The unified imaging model

The unified model is introduced by Geyer and Daniilidis [36], in the context of catadioptric cameras (a modified version is also proposed by Barreto et al. [37]). It is later shown by Ying and Hu [38] (as they replaced the sphere with a general quadric of revolution [15]), that many fisheye lens cameras can also be described using this model (however according to Scaramuzza [3], their approach had limited accuracy).
Geyer and Daniilidis employed a two-step procedure to model a central catadioptric camera:

- In step one (spherical projection) a scene point in a 3-d space is projected on the surface of the unit sphere via the origin of the sphere (also named the viewing sphere [38]). Note that the origin of the sphere represents the single effective viewpoint.\textsuperscript{14}
- In step two, the point on the viewing sphere is re-projected onto a 2-d projective plane via some other viewpoint. This new viewpoint is located at distance $\epsilon$ along an axis, above the origin of the sphere.

Geyer and Daniilidis model is important because it is employed as a stepping stone to other unified/ generic models. As explained by Kannala et al. [23] although central catadioptric cameras and fisheye cameras have a different physical construction, they are not too different from the viewpoint of mathematical modeling. Kannala et al. also propose a generic camera model, for wide-angle/ fisheye lenses, where it considers the relation between distorted radial distances and incidence angles between camera rays and the optical axis.

The calibration procedure in “Robotics, Vision and Control” [12] is expressed by Hansen et al. The procedure, estimates the distance from the 2-d imaging plane and a viewpoint above the origin of the unit sphere. Other examples, taking advantage of the projection model of Geyer and Daniilidis, are the toolbox of Mei [17] and the toolbox of

\textsuperscript{14} Emphasis on why the single effective viewpoint is so desirable.
Barreto [39], [37]. Finally the toolbox of Scaramuzza utilizes a unified Taylor model (as it estimates a polynomial function $f(\rho)$) for catadioptric and fisheye lens cameras.

Equation 2.17 derived by Scaramuzza [22], is Geyer and Daniilidis unified model for central catadioptric cameras. Scaramuzza version of the unified/generic model is capable of defining wide-angle/fisheye lenses and it is employed to model the fisheye lens cameras in this work.

$$X_s = g(m) \propto \begin{bmatrix} \frac{x_m}{1 - \epsilon + \sqrt{1 + (1 - \epsilon^2)(x_m^2 + y_m^2 + 1)}} \\ \frac{y_m}{1 - \epsilon + \sqrt{1 + (1 - \epsilon^2)(x_m^2 + y_m^2 + 1)}} \\ \frac{x_m^2 + y_m^2 + 1}{1 - \epsilon + \sqrt{1 + (1 - \epsilon^2)(x_m^2 + y_m^2 + 1)}} \end{bmatrix} \tag{2.17}$$

Assume that the axis of symmetry of the mirror is perfectly aligned with the optical axis of the camera. Also the camera and the mirror reference frames, differ only by a translation along the z-axis. In Equation 2.17 if $X = (x, y, z)$ is a scene point in the mirror reference frame, centered in $C$, $X_s = (x_s, y_s, z_s)$ is the projection of the scene point on the unit sphere. The point coordinates are then changed to a new reference frame centered in $C_e = (0, 0, -\epsilon)$; therefore $X_e = (x_s, y_s, (z_s + \epsilon))$. Variable $\epsilon$ is distinct for different types of mirrors and ranges between zero (for planar mirror) and one (for parabolic mirror)$^{15}$. $X_e$ is then projected onto the normalized image plane distance 1 from $C_e$ accordingly: $\tilde{m} = (x_m, y_m, 1) = \left( \frac{x_s}{x_s + \epsilon}, \frac{y_s}{y_s + \epsilon}, 1 \right) = g^{-1}(X_s)$.

Finally, the point $\tilde{m}$ is mapped to the camera image point $\tilde{x} = (u, v, 1)$ through the intrinsic matrix $K$.

$^{15}$ The correct value of $\epsilon$ can be obtained knowing the distance between the foci of the conic and the latus rectum; as shown in [3].
\( \bar{x} = K \bar{m} \)  \hspace{1cm} (2.18)

In Equation 2.17 the symbol \( \propto \) indicates that function \( g \) is proportional to the quantity on the right-hand side. To obtain the normalization factor, it is sufficient to normalize \( g(m) \) onto the unit sphere (more details can be found in [3]).

2.4 Camera calibration

Before constructing a perspective image from an omni-directional image and using algorithms (including POSIT) available for perspective projection, the geometry of the fisheye lens camera must be accurately modeled.

- Camera calibration can be performed by observing a 2-d planar calibration pattern\(^{16}\) (e.g. a chessboard or a circular-dot).
- Camera calibration can be performed by observing a 3-d object (this method usually includes 2 or 3 planes orthogonal to each other).
- Camera calibration can be performed by moving a camera in a static scene, also known as Self-calibration (No calibration object is needed).

When using 3-d calibration objects, fewer images of the object, and even in some cases only one image is required to calibrate the camera. However the advantage of using fewer images is lessened, since multiple images of the object are needed. Note that multiple images are required in order to increase accuracy of the calibration.

\(^{16}\) We employed 20-30 good snapshots of the calibrate plane in different positions to define our wide-angle/ fisheye lens camera.
process. In the following paper [26], to calibrate their camera, Bakstein and Pajdla use a single image of a 3-d calibration object.

In Self-calibration, the procedure does not provide one with sufficient accuracy to extract metric information (including pose) from the 2-d images the camera captures. An obvious reason to utilize known calibration objects, not available in Self-calibration, is the adverse effect of mistaking curves in the scene as straight lines in the image.

2-d planar calibration pattern

In this technique camera calibration is performed by observing a pattern with known geometry. The 2-d pattern consists of features that can be extracted either manually or automatically (exploiting available vision algorithms). The features in the pattern outline a straight line and the straightness of the lines, in the image, is employed to define the camera (and solve for camera parameters) in camera calibration.

When using the 2-d calibration, multiple images of the plane are required. Influencing number of images of the 2-d calibration plane is the pre-requisite to assure that all the visible areas of the camera is covered. In other words, one needs to move the chessboard (calibration plane) enough between images to obtain a “rich” set of views [11]. Other factors influencing number of images are the math (see [11] for additional details) and the amount of noise present in the image. Having multiple
images of the calibration board is also necessary for precise detection of the center of the fisheye camera or the center of distortion.

The 2-d calibration pattern is the most common approach to calibrate a camera. Other benefits of using a 2-d calibration plane, are that the math behind the calibration process is simpler, and it is very easy to construct the 2-d pattern; allowing users to move from the laboratory setting to the real world.¹⁷

![Sampled images of the calibration plane](image)

Figure 2.6. Sampled images of the calibration plane

---

¹⁷ According to [11], some calibration methods rely on 3-d objects (e.g., a box covered with markers), but flat chessboard patterns are much easier to deal with; it is difficult to make (and to store and distribute) precise 3-d calibration objects. OpenCV thus opts for using multiple views of a planar object (a chessboard or a circle-dot) rather than one view of a specially constructed 3-d object.
Optimization

A full-scale non-linear optimization of camera parameters is needed when calibration accuracy is important, regardless of the camera model [23]. Note that the non-linear optimization can be time consuming or computationally expensive (however this is less of an issue now that the processing power of personal computers is immense).

Center of distortion

During calibration, in majority of calibration toolboxes, the optical center of distortion is measured. A general approach to determine center of distortion, explained in [30], is as follows: By using a set of images taken of the calibration board the curvature of each line is estimated. Intersections of lines, with minimum curvature in orthogonal direction, indicate the approximate center of distortion. Several approximate centers of distortion have been averaged out to obtain better/ reliable estimate. Center of distortion is measured in all directions symmetrically and expressed in terms of radial distance.

Distortion correction/ un-distortion

The radial distortion of a wide-angle lens can be optically corrected using a combination of compensating lenses, as explained by Hughes et al. [35]. However according to Bogner [40] the amount of distortion that can be corrected in this manner
is physically limited. Therefore for a fisheye lens camera, some form of post-processing is required to correct the radial distortion.

In distortion correction, pixels of an image captured via a fisheye lens camera (the distorted image), are re-mapped creating a perspective image. To un-distort and create a perspective image first the fisheye lens camera has to be defined and its distortion characterized. Notice that un-distortion is performed in 2-d pixel space or coordinate.

2-d geometric image transforms are typically performed in reverse order (from destination/ target to source) as to avoid sampling artifacts (holes and multiple hits). If an image from source to destination is generated, depending on the transform, there will be pixel locations in the destination image that are not hit at all (creating holes) and there will be other pixel locations that are hit multiple times by the source pixels (consequently losing some image content), as explained in [10]. To generate an image from destination to source one starts with a blank slate for the target image. For every pixel on the blank slate one determines what source pixel to use. This process happens according to the reversed mapping function.

In distortion correction if the forward-model is employed to un-distort images, the resultant images will contain many vacant pixels. Interpolation methods can be utilized to overcome this issue [19]. Alternatively, instead of using the forward model and performing interpolation, back-mapping (or inverse-mapping) may be applied to overcome the problem of vacant pixels [35].
Forward\textsuperscript{18} and back-projections are an important property of camera models. Strum et al. [15] explains how some models have a direct expression for forward-projections, others for back-projections, and some work easily both ways. This is important since forward and back-projections are convenient for different tasks: forward-projection for distortion correction of images and bundle adjustment, and back-projection for minimal methods for various structure-from-motion tasks, such as pose and motion estimation.

\textsuperscript{18} The mapping function or the forward transform, maps form source image to destination (also known as the wrap module in some literature).
2.5 Metric information from 2-d images

2.5.1 Measuring pose

A 3-d Cartesian coordinate system is employed to represent objects their pose is about to be measured. A coordinate system also represents the camera imaging the objects. In the POSIT algorithm, orientation (Rotation matrix 2.19) and position (Translation vector 2.20) of the object (together constituting the pose of an object, Equation 2.21) with respect to the camera coordinate frame are computed.

The POSIT algorithm utilizes a single image of an object to measure its pose. The algorithm requires a rigid object with known structure. The rigid object in space is defined by set of points with known arrangements relative to the origin of the object’s coordinate frame\(^{19}\).

\[ R - \text{Rotation matrix:} \]

\[
\begin{pmatrix}
R_{11} & R_{12} & R_{13} \\
R_{21} & R_{22} & R_{23} \\
R_{31} & R_{32} & R_{33}
\end{pmatrix}
\]  

(2.19)

\[ t - \text{Translation vector:} \]

\[
\begin{pmatrix}
t_x \\
t_y \\
t_z
\end{pmatrix}
\]  

(2.20)

\(^{19}\) In computer vision and robotics an object in 3-d space (our object of interest) can be defined by its own Cartesian coordinate frame. Think of a square cube; one vertex is the origin of the frame and location of all other vertices is known in relation to the origin.
The POSIT algorithm is classified as Perspective-n-Point problem (PnP for short) or non-coplanar P4P problem, to be specific. As a set of points on the object (minimum 4 non-coplanar points) are related in the scene with their corresponding coordinates in the image plane\(^{20}\) using the camera intrinsic parameters.

The pose \(P\) of a 3-d object is a combination of its orientation \(R\) and its position \(t\) relative to the camera (12 unknowns to solve for). The rotation matrix and the translation vector are combined to create a 3 x 4 matrix known as the transformation matrix:

\[
P = [R|t]
\]  

(2.21)

In projective geometry the transformation matrix must be supplied in Homogeneous coordinates (making it a 4 x 4 matrix). Homogeneous geometry allows points and lines at infinity, which is a concept that does not exist in Euclidean geometry. In addition, to represent an affine transformation, translation can be expressed with matrix multiplication (see Appendix A).

\(^{20}\) \((u_i, v_i), (X_i, Y_i, Z_i)\) For additional details refer to [12].
3. Methodology

To measure the pose of an object with known dimensions using a fisheye camera, a Point Grey Chameleon camera with a USB interface is used. The board-level camera is equipped with a high quality fisheye lens. The field of view of the fisheye lens, manufactured by OmniTech Robotics, is 190°. The lens is attached to the camera with a micro lens mount (12 mm x 0.5 mm) and is small in size (explained in the Background, the small size of the lens allows for better approximation of the single effective viewpoint property).

As it is already established, the pinhole camera model cannot be applied to define an omni-directional camera which includes the fisheye lens mentioned above. In this the general Taylor polynomial model, developed by D. Scaramuzza [22], is selected to define the fisheye lens camera. After the fisheye lens camera has been defined, images captured by the camera are converted to perspective images. The POSIT algorithm [41] can now be utilized to measure the pose of objects with known dimensions in the scene.

All experiments have been conducted in an indoor environment, outfitted with a 6DOF object tracking system called the OptiTrack. The system, manufactured by Natural Point, includes 6 motion capture cameras and a Tracker software. The Tracker software is engineered to track objects (3-d location of markers attached to objects) in 6DOF with exact precision; that supports real-time and offline workflows [42]. According to
Natural Point, The 3-d location of markers can be resolved with millimeter accuracy and resolution depending on capture volume size and camera configuration.

In the experiments, depth is defined as the distance of an object to that of the camera and displacement as movement of the object in the scene, in relation to the camera. In order to evaluate the performance of the POSIT algorithm, using images captured via the fisheye lens camera, a measuring tape is utilized to measure displacement. To measure depth the OptiTrack system is employed. Although employing a measuring tape in this case is feasible, the need to move around the object to cover all the visible areas of the camera, while the camera is stationary, makes a measuring tape inconvenient to use.

In Experiment 1 different configurations of cameras and wide-angle/ fisheye lenses are tested. The objective in this experiment is to maximize field of view while a quality perspective image is created. Before extending the field of view of the camera via a fisheye lens, in Experiment 2 the focus is on using a perspective camera to run the POSIT algorithm. In Experiment 3 we extend the camera’s field of view and perform the POSIT algorithm. In Experiments 2 and 3, after measuring the pose, the Euclidean distances between objects in the scene are calculated.

Not enough measurements are available, in Experiment 3, to fully analyze the performance of the POSIT algorithm, after enhancing field of view. Therefore in Experiment 4 many more measurements, under better circumstances, are collected. In this experiment displacement of the object is constant at 90 centimeters. In addition,
the 4 non-coplanar points (corners of the object its pose is about to be measured) are illuminated for better 2-d image extraction.

Note that the shape of the object in the scene, in Experiments 2, 3, and 4, is cubic. The cubic shape simplifies the position analysis. A Cartesian coordinate system represents the position of an object in 3-d space, and each object in space is defined by its own coordinate frame. Consider one vertex (or corner) of the cube as the origin of the coordinate frame. The sides of the cube (3 meeting at each vertex), connected to the origin, are considered the axis of the object frame.

The camera as well is defined by a Cartesian coordinate system. The principal point of the camera is the origin and the optical axis is one of its axes. The other two axes of the camera’s coordinate frame are defined by the right-hand rule.

In Experiment 5, the field of view is enhanced as 3 conventional perspective cameras are combined. The calibration of a multi-camera imaging system, involves computing the calibration matrix and the distortion parameters for all 3 cameras, as well as the position and the orientation of each camera with respect to camera 1. B. Li’s multi-camera calibration toolbox [43] is utilized to calibrate the 3-camera imaging system. After calibration, H. C. Longuet-Higgins’ algorithm [44] is exploited to calculate position of 3-d points in the scene. Subsequently the distances between the points in the scene and the distances between the points and camera 1 are measured.

The idea behind Experiment 5 is to enhance the field of view of an imaging system using some other technique. The 3-d measurements performed via this new
method are then compared with the measurements conducted using the fisheye camera.
3.1 Fisheye lens camera calibration

When a fisheye camera is to be utilized to make 3-d measurements, the first step is to define the geometry of the camera. That includes estimating the camera’s intrinsic values and distortion parameters during the calibration process. Different camera models are introduced in the Background section of our thesis and the general Taylor polynomial model developed by D. Scaramuzza [22] has been selected and utilized to define the fisheye lens camera in this work. The work of D. Scaramuzza comes with a toolbox [45] and it is inspired by the popular J. Bouguet’s calibration toolbox [46]. Unlike J. Bouguet’s toolbox, this toolbox can define wide-angle/ fisheye lens cameras.

D. Scaramuzza toolbox starts by asking users to collect images (shown at different positions and orientations) of a 2-d calibration pattern with known geometry (the calibration chessboard). The toolbox then extracts the corner points of the chessboard automatically. After a desired order is selected for the polynomial\(^{21}\), the calibration takes place. Following the calibration one can convert an omni-directional image to a perspective image.

Figure 3.1 (Top) has been captured via a miniature, low-cost, fisheye lens with unknown and undefined projection function. Figure 3.1 (Bottom) is the same image after it has been converted to a perspective image, via D. Scaramuzza’s unified camera model. The camera model treats the imaging system as a unique compact system;

\(^{21}\) Experiments conducted by D. Scaramuzza shows that a polynomial of order 4 gives the best results.
therefore it does not care if a mirror or a fisheye lens in combination with a camera is being used [45].

Figure 3.1. Distortion correction

Image of cubes captured via a fisheye lens camera before (Top) and after the image has been converted to a perspective image (Bottom). (Original in color)
Equation 3.1 is proposed by D. Scaramuzza et al. and the main difference with Geyer and Daniilidis’ unified model (Equation 2.17) lies in the choice of the function $g$; where the function is applied to overcome the lack of knowledge of a parametric model for fisheye cameras [3]. In Equation 3.1 $\rho = \sqrt{x_m^2 + y_m^2}$ is the distance from the center of the omni-directional image in pixel units.

$$X_s = g(m) \propto \begin{bmatrix} x_m \\ y_m \\ a_0 + a_1 \rho + a_2 \rho^2 + a_3 \rho^3 + a_4 \rho^4 + \ldots \end{bmatrix}$$  \quad (3.1)$$

In the general Taylor polynomial model, lens distortion has already been integrated into the projection function; hence no distortion parameters are required. Furthermore, similar to other camera calibration practices additional parameters are added for misalignment and non-orthogonality of the image plane with respect to the optical axis.

The intrinsic parameters of the fisheye lens camera (Appendix C), derived by the calibration toolbox, cannot be applied in its current format by the POSIT algorithm (or other perspective algorithms). Equation 3.2 is utilized after the calibration process to create the typical calibration matrix [47]:

$$K = \begin{pmatrix} \frac{i_w}{s f} & 0 & i_{wc} \\ 0 & \frac{i_w}{s f} & i_{hc} \\ 0 & 0 & 1 \end{pmatrix}$$  \quad (3.2)$$

In the matrix above, $s f$ is the zoom factor specified by the user during the perspective image creation. Borders of the perspective image usually contain artifacts;
$sf$ can be applied to adjust the size of the borders, as explained by Lee et al. [47]. In the matrix above $i_w$ is the width of the image, and $(i_{wc}, i_{hc})$ represents the coordinates of the center of the image (also represented by $(u_0, v_0)$).
3.2 Multiple-camera system calibration

To calibrate our 3-camera imaging system, in Experiment 5, we employed B. Li’s camera calibration toolbox [43]. The calibration toolbox calculates the camera’s matrix and distortion parameters for all 3 cameras (intrinsic parameters), as well as the position and orientation of each camera with respect to camera 1 (extrinsic parameters), Appendix D.

Similar to the previous calibration toolbox, this toolbox employs a 2-d calibration pattern. However this pattern is unique as it uses several noise images to compose a calibration pattern in accordance with the mechanism of SIFT/ SURF (Figure 3.2). Benefits of using such pattern, according to its author, is that it contains many more detectable features on multiple scales. In addition, before calibration, the whole pattern has already been recognized by the toolbox; therefore the pattern can be recognized and localized even if it is captured partially.

Figure 3.2. B. Li 2-d calibration pattern
The toolbox’s other contribution is that no overlapping field of view, for the multi-camera imaging system is assumed. As a result in order to calculate extrinsic parameters (position and orientation of neighboring cameras) the toolbox only requires the neighboring camera, to observe parts of the calibration pattern (the observed parts do not need to overlap at all). The downside of this approach is that neighboring cameras still have to observe parts of the calibration pattern at the same time. Consequently if neighboring cameras are far apart then a fairly large calibration object is required.
3.3 POSIT algorithm

The POSIT (Pose from Orthography and Scaling with Iteration) algorithm, adequately explained in [48], [49], and [41], is exploited to measure the pose of objects in the scene with respect to the camera. The requirements to execute the algorithm are that the 3-d objects are entirely in the camera’s field of view and have a nominal size in the image (not too distant). It is also required to know the camera’s intrinsic parameters and dimensions of the objects.

Benefits of using the POSIT algorithm is that it requires only 1 camera to measure the pose of an object. Also POSIT is simple to implement and as an iterative pose estimating algorithm, POSIT can achieve superior accuracy provided that it converges to a solution; excellent precision of the approach comes at the price of calculation time [2].
Figure 3.3. Schematic diagram of the POSIT algorithm
To run the POSIT algorithm, one first needs to extract at least 4 non-coplanar points on the image (Figure 3.5) and match the extracted points with its corresponding model points (Figure 3.4). It is necessary for at least one of the 4 points to be non-coplanar; otherwise the equation for pose transformation can’t be solved. Now POSIT is a two part procedure [41]:

The pose from Orthography and Scaling (POS) assumes weak-perspective camera model (also known as scaled orthographic projection), allowing the use of a linear closed form solution to estimate the object’s 3-d pose based on scaling. Having estimated the rotation matrix and the translation vector of the object coordinate frame with respect to the camera, the results are not very accurate.

For the second part, POS with Iteration (weak-perspective image points are expressed in terms of strong perspective points) applies POS to the approximate pose found in the previous step, so that it can compute better scaled orthographic projections of the feature points. It takes four to five iterations for the process to converge to a more accurate object pose.
Figure 3.4. The four non-coplanar points
(Origin on color)

Figure 3.5. The four non-coplanar Image points
(Origin on color)
Figure 3.4 is a known object, where \( S \) is the distance from the reference point, which its pose is about to be measured. 4 non-coplanar points are selected with one point representing the origin of the object’s frame. Figure 3.5 displays the corresponding non-coplanar Image points. Figure 3.6 is a summary of steps carried out to employ a fisheye lens camera to measure pose of objects in the scene.
Figure 3.6. Proposed approach to enhance the camera’s field of view.
3.4 Position from corresponding images

Assume the imaging system is comprised of 2 perspective cameras; camera 1 and camera 2. The position and orientation of camera 2 with respect to camera 1 as well as the focal lengths \((f_x, f_y)\) and coordinates of the principal point \((u_0, v_0)\) in pixel units for both cameras are available after camera calibration. If a point is seen by both cameras simultaneously, its position in 3-d space can be measured with respect to camera 1, in 3 steps, using H. C. Longuet-Higgins algorithm [44].

Step 1: In order to compute the distance of a point in 3-d space with respect to camera 1; the position/ coordinate of the point is extracted in the 2-d image space (for both images captured via camera 1 and 2):

\[
\begin{align*}
\text{position of the point in 3d w.r.t. camera 1} &= [X \quad Y \quad Z]^T \\
\text{image point 1} &= \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}, \quad \text{image point 2} = \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}
\end{align*}
\]

(3.3)
Step 2: By using focal lengths and coordinates of the principal point in pixel units, the position/coordinate of the point in the 2-d image space is normalized according to Equation 3.4:

\[
\begin{align*}
\chi_{\text{norm}} &= \frac{(x - u_0)}{f_x} \\
y_{\text{norm}} &= \frac{(y - v_0)}{f_y}
\end{align*}
\] (3.4)

Step 3: H. C. Longuet-Higgins’ algorithm is utilized to measure the distance of the point in 3-d space with respect to camera 1, using Equation 3.5:

\[
\begin{align*}
\begin{pmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{pmatrix} &= \begin{pmatrix} R_1 \\ R_2 \\ R_3 \end{pmatrix} \\
Z &= \frac{(R_1 - x_2^{\text{norm}} R_3) \cdot t}{(R_1 - x_2^{\text{norm}} R_3) \cdot \left(\frac{x_3^{\text{norm}}}{y_3^{\text{norm}}} \right)} \\
\begin{pmatrix} X \\ Y \end{pmatrix} &= Z \begin{pmatrix} x_1^{\text{norm}} \\ y_1^{\text{norm}} \end{pmatrix}
\end{align*}
\] (3.5)
4. Experiments and analysis

4.1 Experiment 1: Field of view

In this experiment the extent to which segments of an image can be kept, is examined after generating a perspective image. This experiment shifts the focus away from omni-directional camera models and the accuracy created as a result of different calibration processes, and instead focuses on limitations that are inherent in such imaging systems.

When using a fisheye lens camera, to perform 3-d measurements, it is essential to generate top-quality perspective images. Moving away from the center of distortion, the overall resolution of the image captured by means of the fisheye lens camera decreases. As resolution reduces toward the peripheries of the image, not enough image points are available to construct a quality perspective image. This restriction generates difficulty extracting the corners of an object of interest, on the peripheries of the image, in order to perform the POSIT algorithm.

Another reason the corners of an image, generated from an omni-directional image, are not suitable to perform 3-d measurements, is that features (e.g. the cube its pose is about to be measured) in the peripheries appear “stretched”. To analyze this occurrence consider the unified imaging model, where in Step 2 an image point on the viewing sphere is re-projected onto a 2-d projective plane. Geometrically it is difficult to re-project an entire hemisphere (180°) onto a 2-d projective plane. As a result for
perspective images generated from omni-directional images, portions of the image cannot be used, thus cropped. Unfortunately this process reduces the effective field of view of the imaging system.

Another factor influencing the field of view of the imaging system, as displayed in Appendix B, is the relationship/combination of the fisheye lens and the image sensor. In order to achieve maximum field of view, the image sensor (green rectangle shown in Figure 4.1 (Left)) must be larger than the image circle (red circle shown in Figure 4.1 (Left)). This combination creates a full-circular image. However in this scenario, fractions of the image sensor are not exploited.

On the other hand to achieve maximum resolution, the image sensor needs to be exactly equal to the largest useable rectangle (green rectangle shown in Figure 4.1 (Right)). This combination creates a full-frame image\textsuperscript{22} and it is the ideal setup. Under this condition all active areas of the image sensor are employed to capture the useable section of the image circle, producing the highest spatial resolution possible.

Also note that when the sensor is equal to the largest useable rectangle, in the diagonal direction, the field of view is $180^\circ$; that is of course if the lens captures a full hemisphere ($180^\circ$). So the field of view in the horizontal and vertical direction is less than $180^\circ$.

\textsuperscript{22} Also a full-frame image is created if the circular image is much bigger than the image sensor. Now sections of the scene, seen by the lens, cannot be recorded by the sensor, hence field of view is reduced.
It should be mentioned, that the ideal setup may not be achievable and that it is contingent to the size of the circular image produced by the lens, the sensor format, and the focal length of our imaging system.
Figure 4.1. The image sensor and circular image
180° diagonal field of view shown (Right). (Origin in color)

(a)  
(b)  

Figure 4.2. From full-circular image to a perspective image
(Origin in color)
Assuming that the fisheye lens camera produces a full-circular image, to create a perspective image from an omni-directional image, the following steps are taken: As the entire image circle, produced by the fisheye lens, is captured by the image sensor (the red circle shown in Figure 4.2 (a)), sections of the sensor are left out, therefore unfilled. The vacant sections of the image sensor along with the space between the image circle and the largest useable rectangle, which cannot be utilize, are removed. In this stage only the largest useable rectangle remains (the green rectangle shown in Figure 4.2 (a)), creating a full-frame image (Figure 4.2 (b)). It is optional to trim the full-frame image to eliminate stretched/ low resolution peripheries at this stage, however trimming the edges of the image can equally happen after the perspective image has been generated.

In Figure 4.2 (c) the full-frame image has been corrected for distortion and so a perspective image is created. Note, to generate the perspective image, the focal length has been selected to achieve maximum field of view (See the Unified/ Generic models and the Fisheye lens camera calibration in the Methodology section for details on selecting the focal lengths).

The generated perspective image has a pin-cushion shape. The “arms” are of no use and cannot be exploited in pose estimation; therefore they are removed. After removing the “arms” we are left with our desired top-quality perspective image (Figure 4.2 (d)). The generated perspective image has a smaller field of view than the original.
full-circular image. However the field of view is larger than any conventional perspective camera.

In most circumstances, when dealing with vision algorithms, the focal lengths are needed in pixel units. To calculate focal length in pixel units we need to know the resolution of the image sensor. The focal length in pixel units is calculated accordingly:

\[
Focal \ length \ in \ pixels = (image \ width \ in \ pixels) \times \frac{\text{focal length in mm}}{\text{CCD width in mm}} \tag{4.1}
\]

The field of view of a camera is a function of its focal length \(f\) and the dimensions of the camera chip \(W_{\rho w} \times H_{\rho h}\) (image sensor) [12]. In Equation 4.2 \(\rho w\)
and \( p_w \) are the width and the height of each pixel. The field of view of an imaging system can be calculated accordingly:

\[
\theta_h = 2 \tan^{-1} \left( \frac{W_{pw}}{2f} \right)
\]

\[
\theta_v = 2 \tan^{-1} \left( \frac{H_{ph}}{2f} \right)
\]

(4.2)

The following experiment examines to what extend the field of view has improved by means of different configurations of image sensors and wide-angle/fisheye lenses.
4.1.1 Configuration 1

Table 4.1. High quality camera

<table>
<thead>
<tr>
<th>Point Grey Chameleon (CMLN-13S2M) Camera</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Sensor Size (format/ type) in inch:</td>
</tr>
<tr>
<td>Image Sensor Dimensions (D x W x H) in millimeters:</td>
</tr>
<tr>
<td>Maximum Resolution in pixels:</td>
</tr>
<tr>
<td>Description:</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

Table 4.2. High quality fisheye lens

<table>
<thead>
<tr>
<th>Omni tech Robotics (ORIFL 190-3) Lens</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field of View (circular) in degrees:</td>
</tr>
<tr>
<td>Focal length in millimeters:</td>
</tr>
<tr>
<td>Description:</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Focal Length in pixels for CMLN-13S2M camera:</td>
</tr>
</tbody>
</table>

This is the main configuration and it is employed in Experiment 3 and 4 to estimate pose. The system includes a high performance digital camera and a high quality fisheye lens that provides 190° field of view. Designed and built exclusively by and for OmniTech Robotics, the ORIFL190-3 is optimized for small size and high image quality [50]. The camera is a Point Grey Chameleon (CMLN-13S2M0) and its ICX445AKA image sensor has a 1296 x 964 pixels resolution [51].
In this setup the image sensor is bigger than the image circle; therefore portions of the image sensor are not utilized. With the goal to achieve the maximum field of view while creating a top-quality perspective image, different zoom factors are examined. The selected zoom factor is set at 2.8, while the dimensions of the image sensor are 550 x 520 pixels resolution. The focal length of the virtual perspective image generated is calculated at 197 pixel units. Table 4.3 demonstrates the measured field of view of the fisheye camera.

Table 4.3. The approximate fisheye Field Of View
(Diagonal, Horizontal, and Vertical)

<table>
<thead>
<tr>
<th>FOV(D)</th>
<th>125.1°</th>
</tr>
</thead>
<tbody>
<tr>
<td>FOV(H)</td>
<td>108.9°</td>
</tr>
<tr>
<td>FOV(V)</td>
<td>105.7°</td>
</tr>
</tbody>
</table>

To perform a comparison, the digital camera is paired with a conventional perspective lens. The perspective lens has no spec sheet, so its optimal focal length is measured as one calibrates the camera (focal length is 1016 pixel units at 1280 x 960 pixels resolution). Table 4.4 is the measured field of view of the perspective camera.
Table 4.4. The approximate perspective FOV

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>FOV(D)</td>
<td>76.4°</td>
</tr>
<tr>
<td>FOV(H)</td>
<td>64.4°</td>
</tr>
<tr>
<td>FOV(V)</td>
<td>50.6°</td>
</tr>
</tbody>
</table>

As displayed by Tables 4.3 and 4.4, when the fisheye lens is applied, the horizontal field of view has increased\(^{23}\) by approximately 70% while the overall field of view (diagonal) has increased by approximately 64%.

Unfortunately when the fisheye lens is mounted on the digital camera, big portions of the image sensor are not utilized. If bigger portions of the image sensor could theoretically be exploited then spatial resolution of the image would be higher. Improved resolution is quite practical when extracting the corners of an object in order to measure pose, exercising the POSIT algorithm.

---

\(^{23}\) Percent change = ((new FOV − old FOV) / old FOV) \times 100
4.1.2 Configuration 2

Table 4.5. Wide-angle lens

<table>
<thead>
<tr>
<th>Field of View in degrees (when used with 1/3&quot; camera):</th>
<th>100°(H) x 74°(V)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Focal Length in millimeters:</td>
<td>variable (2.8 - 8)</td>
</tr>
<tr>
<td>Description:</td>
<td>• wide angle lens</td>
</tr>
<tr>
<td>Focal Length in pixels for CMLN-13S2M camera:</td>
<td>(1296 x 2.8mm) / 4.8mm = 756</td>
</tr>
<tr>
<td></td>
<td>(1296 x 8mm) / 4.8mm = 2160</td>
</tr>
</tbody>
</table>

In this configuration the Point Grey digital camera has been paired with a wide-angle lens; manufactured by Fujifilm optical lenses (Fujinon). The wide-angle lens has a variable focal length and is capable of high image quality with optical performance supporting 3 megapixels [52]. At 2.8 millimeters the focal length of the camera should achieve a horizontal field of view of 100° (according to the spec sheet). As displayed in Table 4.5 the focal length can be anywhere from 756 - 2160 pixel units when the lens is paired with the Point Grey digital camera.

In order to attempt to capture the sharpest image possible, the focal length is selected. During calibration the focal length is revealed to be 1328 pixel units. Therefore the field of view of the camera is measured accordingly:
Table 4.6. The approximate wide angle FOV

<table>
<thead>
<tr>
<th>FOV</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FOV(D)</td>
<td>62.1°</td>
</tr>
<tr>
<td>FOV(H)</td>
<td>51.4°</td>
</tr>
<tr>
<td>FOV(V)</td>
<td>39.7°</td>
</tr>
</tbody>
</table>

In this configuration the field of view of the wide-angle lens is disappointing. In this setup the image circle is much bigger than the image sensor, hence large segments of the scene, seen by the lens, cannot be captured by the image sensor. At shorter focal lengths, where the field of view is greater (according to the spec sheet), unfortunately the captured image is blurred, thus not functional.
4.1.3 Configuration 3

Table 4.7. Cheap off the shelf webcam

<table>
<thead>
<tr>
<th>Logitech (C210) Camera</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Image Sensor Size</strong> (format/ type) in inch:</td>
</tr>
<tr>
<td><strong>Image Sensor Dimensions</strong> (D x W x H) in millimeters:</td>
</tr>
<tr>
<td><strong>Maximum Resolution</strong> in pixels</td>
</tr>
</tbody>
</table>
| **Description:** | • cheap webcam  
| | • color |

Table 4.8. Unknown/ undefined fisheye lens

<table>
<thead>
<tr>
<th>Kodak Lens</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Field of View</strong> (circular) in degrees:</td>
</tr>
<tr>
<td><strong>Focal Length</strong> in millimeters:</td>
</tr>
<tr>
<td><strong>Description:</strong></td>
</tr>
</tbody>
</table>

Previously the intend to calibrate miniature, low-cost, fisheye lens cameras with unknown/ undefined projection functions has been expressed. This configuration combines a cheap off-the-shelf webcam and a very basic fisheye lens. The fisheye lens manufactured by Kodak has no spec sheet and is intended to be mounted on a mobile phone (mostly smart phones).

In this setup the zoom factor, with the goal to create a top-quality perspective image has been set at 2.0. Using the zoom factor and the dimensions of the image sensor (640 x 480 pixel units) the focal length of the virtual perspective image is calculated at 320 pixel units. Field of view of the camera is measured accordingly:
Table 4.9. The approximate cheap configuration FOV

<table>
<thead>
<tr>
<th>FOV</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FOV(D)</td>
<td>102.7°</td>
</tr>
<tr>
<td>FOV(H)</td>
<td>90.0°</td>
</tr>
<tr>
<td>FOV(V)</td>
<td>73.7°</td>
</tr>
</tbody>
</table>

Surprisingly in this configuration, the field of view is sound. Unlike Configuration 1, the size of the image circle is greater than the size of the imager. Therefore there is no need to crop the image to create a full-frame image (from a full-circular image). However after generating a perspective image the “arms” of the pin-cushion image must be removed.
4.2 Experiment 2: The POSIT algorithm

Before extending the camera’s field of view, we simplify matters and focus only on pose estimation. In this experiment, a perspective camera is employed to run the POSIT algorithm. The purpose of this experiment is to analyze the effectiveness of the POSIT code developed, by calculating the distances between the cubes in the scene, as shown in Figure 4.4 (Top).

The Logitech C210 camera, with 640 x 480 pixels resolution, is applied to capture the image of the cubes. In step 1 the perspective camera is calibrated. To calibrate the camera, OpenCV24 [53] programming functions are exploited. Detailed adequately in [11], the OpenCV (library) approach to camera intrinsics is derived by Heikkila and Silven [27]. The algorithm applied by OpenCV to solve for the focal length and the offsets (principal point and center of the imaging plane) is based on Zhang’s work [54] and to solve distortion parameters, Brown’s method is utilized [28].

To calibrate the camera, approximately 10 images of a 7 x 5 chessboard (2-d) have been captured. After operating the OpenCV camera calibration program, the software produces a camera matrix (matrix of intrinsic parameters) and a distortion vector (coefficient).

24 OpenCV is a library of computer vision programming functions and it has a C++ interface [53].
Figure 4.4. Calculate perspective distance between cubes  
Accounting for distortion then running POSIT (Bottom) (Origin in color)
In step 2 before measuring the pose of the cubes, in the scene, the raw perspective image has been un-distorted. Pay attention to the edges of the image (Figure 4.4 Bottom). This is how the image appears after accounting for distortion (radial and tangential). As it can be seen by the image, there is not much distortion to account for.

In step 3 the corners of the cubes (using the undistorted image) are extracted manually (defined a callback function for a mouse click event in OpenCV). The extracted corners represent 4 non-coplanar points that are needed to calculate the pose in the POSIT algorithm.

In step 4 the POSIT algorithm is applied to calculate the pose of each cube in the image. The POSIT algorithm is defined by D. F. DeMenthon and L. S. Davis in [41], and it is available in OpenCV (library). A practical example of the algorithm has been written, in C++, by J. Barandiaran [55] and it influences the POSIT code in this work.

In step 5 after calculating the pose of each cube, in the scene, the distances between the origins of each object frame has been computed (using Euclidean distance; Equation 4.3). The white dots, in Figure 4.4 (Top), signify the origin of each object frame.

\[
Euclidean \ distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}
\]  

(4.3)
As demonstrated by Equation 4.3, to calculate the distance between two object frames in the scene, one only needs the translation vectors of each object. Also note that all cubes in the scene share the same surface (cubes are on the same plane and their coordinate axes are either the same or parallel to one another) therefore have the same orientation. The only variables, in this experiment, are the translation vectors which are exploited to calculate distances between cubes in the scene.

In this experiment, different color cubes are put together. In order to measure the distance between them, the POSIT algorithm has been utilized. In Table 4.10, the first set of measurements, colored yellow, is the pose of the yellow cube on the left side of the image (Figure 4.4). The second set of measurements, colored red, is the pose of the red cube in the scene and so on... The distances calculated are all relative to the yellow cube on the left side of the image.
### Table 4.10. Perspective-view and POSIT algorithm

<table>
<thead>
<tr>
<th>ESTIMATED ROTATION</th>
<th>ESTIMATED TRANSLATION</th>
<th>Euclidean Distance (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2798</td>
<td>-0.179715</td>
<td>0.943088</td>
</tr>
<tr>
<td>0.889127</td>
<td>-0.457658</td>
<td>-0.001419</td>
</tr>
<tr>
<td>0.431867</td>
<td>0.838922</td>
<td>0.031737</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ESTIMATED ROTATION</th>
<th>ESTIMATED TRANSLATION</th>
<th>Euclidean Distance (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.206107</td>
<td>-0.14432</td>
<td>0.967828</td>
</tr>
<tr>
<td>0.889316</td>
<td>-0.457209</td>
<td>0.0087499</td>
</tr>
<tr>
<td>0.441237</td>
<td>0.858902</td>
<td>0.0341122</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ESTIMATED ROTATION</th>
<th>ESTIMATED TRANSLATION</th>
<th>Euclidean Distance (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.23020</td>
<td>0.0657403</td>
<td>0.97092</td>
</tr>
<tr>
<td>0.89383</td>
<td>-0.447757</td>
<td>0.0241021</td>
</tr>
<tr>
<td>0.436321</td>
<td>0.873386</td>
<td>0.0443137</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ESTIMATED ROTATION</th>
<th>ESTIMATED TRANSLATION</th>
<th>Euclidean Distance (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.23085</td>
<td>0.114057</td>
<td>0.966282</td>
</tr>
<tr>
<td>0.893403</td>
<td>-0.44907</td>
<td>0.0129327</td>
</tr>
<tr>
<td>0.435403</td>
<td>0.866264</td>
<td>0.0017675</td>
</tr>
</tbody>
</table>
Table 4.10 displays that the POSIT algorithm/ code can be successfully utilized to measure the pose of objects, seen by the perspective camera. In this experiment however, we do not have enough measurements to analyze the accuracy of the POSIT algorithm. Further analysis of the POSIT algorithm, employing a large set of measurements, has been conducted in the next experiments.
4.3 Experiment 3: Fisheye camera and the POSIT algorithm

Figure 4.5. Enhance field of view via fisheye camera

(Origin in color)

In this experiment a perspective image from an omni-directional image is generated, the POSIT algorithm is performed, and the distances between the cubes in the scene are measured. Steps in this experiment are similar to that of the previous experiment, with the main differences being the use of a different camera model to define the fisheye lens camera and the conversion of the omni-directional image to a perspective image.

The toolbox employed to calibrate the fisheye lens camera, computes the intrinsic parameters in order to generate the perspective image. The toolbox is developed by D. Scaramuzza [45]. Camera Configuration 1, introduced in Experiment 1, is utilized in this experiment.

As shown in Figure 4.5 the camera captures images in grayscale. Therefore the cubes, in the scene (3 cubes each separated by 50 centimeters), are numbered. The results, of the experiment, are shown in Table 4.11 and all the measurements, in the table, are according to the cube’s number.
Table 4.11. Fisheye camera and POSIT algorithm

<table>
<thead>
<tr>
<th>ESTIMATED ROTATION (cube 1)</th>
<th>ESTIMATED TRANSLATION (cube 1)</th>
<th>Euclidean Distance (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0976991</td>
<td>-58.0506</td>
<td>0.0976991</td>
</tr>
<tr>
<td>0.0722258</td>
<td>-9.74004</td>
<td>0.0722258</td>
</tr>
<tr>
<td>0.992592</td>
<td>191.93</td>
<td>0.992592</td>
</tr>
<tr>
<td>n/a</td>
<td>n/a</td>
<td>estimated</td>
</tr>
<tr>
<td>0.200848</td>
<td>-0.979607</td>
<td>0.200848</td>
</tr>
<tr>
<td>-0.979607</td>
<td>0.0055566</td>
<td>-0.979607</td>
</tr>
<tr>
<td>0.0055566</td>
<td>0.999839</td>
<td>0.0055566</td>
</tr>
<tr>
<td>n/a</td>
<td>n/a</td>
<td>actual</td>
</tr>
<tr>
<td>0.972751</td>
<td>-9.74004</td>
<td>0.972751</td>
</tr>
<tr>
<td>0.198818</td>
<td>-0.110213</td>
<td>0.198818</td>
</tr>
<tr>
<td>-0.110213</td>
<td>191.93</td>
<td>-0.110213</td>
</tr>
<tr>
<td>ESTIMATED ROTATION (cube 2)</td>
<td>ESTIMATED TRANSLATION (cube 2)</td>
<td>Distance between cubes 1 and 2</td>
</tr>
<tr>
<td>-0.014247</td>
<td>-0.91726</td>
<td>0.014247</td>
</tr>
<tr>
<td>-0.010902</td>
<td>-10.2469</td>
<td>0.010902</td>
</tr>
<tr>
<td>0.999839</td>
<td>201.918</td>
<td>0.999839</td>
</tr>
<tr>
<td>50.0428</td>
<td>estimated</td>
<td>50.0428</td>
</tr>
<tr>
<td>0.239955</td>
<td>-0.970433</td>
<td>0.239955</td>
</tr>
<tr>
<td>-0.970433</td>
<td>0.0261147</td>
<td>-0.970433</td>
</tr>
<tr>
<td>0.0261147</td>
<td>50</td>
<td>0.0261147</td>
</tr>
<tr>
<td>50</td>
<td>actual</td>
<td>50</td>
</tr>
<tr>
<td>0.969992</td>
<td>0.240288</td>
<td>0.969992</td>
</tr>
<tr>
<td>0.240288</td>
<td>0.0164418</td>
<td>0.240288</td>
</tr>
<tr>
<td>0.0164418</td>
<td>100</td>
<td>0.0164418</td>
</tr>
<tr>
<td>ESTIMATED ROTATION (cube 3)</td>
<td>ESTIMATED TRANSLATION (cube 3)</td>
<td>Distance between cubes 1 and 3</td>
</tr>
<tr>
<td>-0.157104</td>
<td>42.3946</td>
<td>-0.157104</td>
</tr>
<tr>
<td>-0.149018</td>
<td>-9.87835</td>
<td>-0.149018</td>
</tr>
<tr>
<td>0.976275</td>
<td>202.767</td>
<td>0.976275</td>
</tr>
<tr>
<td>101.0282</td>
<td>estimated</td>
<td>101.0282</td>
</tr>
<tr>
<td>0.242797</td>
<td>-0.970035</td>
<td>0.242797</td>
</tr>
<tr>
<td>-0.970035</td>
<td>-0.009014</td>
<td>-0.970035</td>
</tr>
<tr>
<td>-0.009014</td>
<td>100</td>
<td>-0.009014</td>
</tr>
<tr>
<td>100</td>
<td>actual</td>
<td>100</td>
</tr>
<tr>
<td>0.948364</td>
<td>0.23562</td>
<td>0.948364</td>
</tr>
<tr>
<td>0.23562</td>
<td>0.18858</td>
<td>0.23562</td>
</tr>
<tr>
<td>0.18858</td>
<td>Distanced between cubes 2 and 3</td>
<td>0.18858</td>
</tr>
<tr>
<td>51.4202</td>
<td>estimated</td>
<td>51.4202</td>
</tr>
<tr>
<td>50</td>
<td>actual</td>
<td>50</td>
</tr>
</tbody>
</table>
Table 4.11 displays that exploiting a fisheye lens to enhance the field of view of a camera and then measuring pose, is a viable option. In this experiment, we do not have enough measurements to analyze the accuracy of the POSIT algorithm when utilizing a fisheye camera. Further analysis of the POSIT algorithm, employing a large set of measurements, has been conducted in the next experiments.
4.4 Experiment 4: Analyze the accuracy of the fisheye camera

The purpose of this experiment is to manage large set of measurements to analyze the accuracy of the POSIT algorithm, when a fisheye camera is utilized. Two positions, separated by 90 centimeters, are marked on a stand. The POSIT algorithm is employed to measure the pose of the cube, with respect to the camera, in position 1. The cube is moved to position 2 and its pose is measured again. Having the pose of the cube in positions 1 and 2, it should be possible to compute the Euclidean distance between the marked positions (displacement).

The pose of the cube, with respect to the camera, can also be applied to measure the distance of the cube from the camera (depth) using Equation 4.4. In this experiment the distance of the cube from the camera is also measured and analyzed.

\[
Distance\ from\ the\ camera = \sqrt{(x)^2 + (y)^2 + (z)^2}
\]  

(4.4)

The lab utilized to conduct the experiments is equipped with the OptiTrack system. The system can determine the position of objects illuminated by IR reflectors. The distance of the cube from the camera can be accurately measured using the OptiTrack system (using a measuring tape is not convenient in this instance). An IR reflector is attached to the origin of the cube’s coordinate frame. Another IR reflector is mounted on the camera. This allows for the OptiTrack system to measure the distance between the cube and the camera.
To better distinguish and extract the corners of the cube in the image, using the POSIT algorithm, LEDs are employed to mark them. Furthermore, image exposure is reduced, as seen in Figure 4.6 (Left-to-Right).

![Figure 4.6. Corner extraction](image)

Extraction of 4 non-coplanar points while reducing exposure (Left-to-Right)

Table 4.12 contains 15 sets of measurements, measured applying a perspective camera. Table 4.13 contains 50 sets of measurements, measured utilizing a fisheye camera. The measurements include displacement, depth, position of the cube in relation to the optical axis (in degrees), and relative error (for displacement and depth).

In Table 4.12 and 4.13, to analyze depth and compute its relative error\(^{25}\) (in percentage), one assumes that the OptiTrack system presents the actual value and the camera supplies the measured value.

\[^{25}\text{Error (\%)} = \left( \frac{|\text{measured value} - \text{actual value}|}{\text{actual value}} \right) \times 100\]
Table 4.12. Measurements using a perspective camera

15 sets of measurements

<table>
<thead>
<tr>
<th>Perspective</th>
<th>OptiTrack (m)</th>
<th>Depth (m) POSIT</th>
<th>Angle (degrees)</th>
<th>Displacement (cm)</th>
<th>Depth Error (%)</th>
<th>Displacement Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>2.564</td>
<td>2.589</td>
<td>2.475</td>
<td>2.498</td>
<td>6.7</td>
<td>14.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>87.66</td>
<td>3.47</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.50</td>
<td>2.60</td>
</tr>
<tr>
<td>M2</td>
<td>2.209</td>
<td>2.383</td>
<td>2.121</td>
<td>2.301</td>
<td>7.8</td>
<td>29.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>88.50</td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.46</td>
<td>1.67</td>
</tr>
<tr>
<td>M3</td>
<td>1.948</td>
<td>2.082</td>
<td>1.871</td>
<td>2.033</td>
<td>15.8</td>
<td>12.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>88.57</td>
<td>3.96</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.35</td>
<td>1.59</td>
</tr>
<tr>
<td>M4</td>
<td>1.174</td>
<td>1.269</td>
<td>1.135</td>
<td>1.232</td>
<td>28.9</td>
<td>23.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>87.56</td>
<td>3.33</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.94</td>
<td>2.71</td>
</tr>
<tr>
<td>M5</td>
<td>3.715</td>
<td>3.783</td>
<td>3.623</td>
<td>3.691</td>
<td>3.4</td>
<td>17.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>87.39</td>
<td>2.47</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.44</td>
<td>2.90</td>
</tr>
<tr>
<td>M6</td>
<td>3.519</td>
<td>3.394</td>
<td>3.381</td>
<td>3.281</td>
<td>25.0</td>
<td>29.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>87.66</td>
<td>3.93</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.32</td>
<td>2.60</td>
</tr>
<tr>
<td>M7</td>
<td>2.585</td>
<td>2.474</td>
<td>2.517</td>
<td>2.386</td>
<td>29.7</td>
<td>9.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>87.54</td>
<td>2.63</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.55</td>
<td>2.73</td>
</tr>
<tr>
<td>M8</td>
<td>3.104</td>
<td>3.268</td>
<td>3.019</td>
<td>3.202</td>
<td>2.6</td>
<td>17.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>88.06</td>
<td>2.74</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.02</td>
<td>2.16</td>
</tr>
<tr>
<td>M9</td>
<td>3.275</td>
<td>3.113</td>
<td>3.176</td>
<td>3.007</td>
<td>10.7</td>
<td>6.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>87.88</td>
<td>3.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.42</td>
<td>2.36</td>
</tr>
<tr>
<td>M10</td>
<td>1.693</td>
<td>1.908</td>
<td>1.645</td>
<td>1.878</td>
<td>20.9</td>
<td>7.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>88.19</td>
<td>2.86</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.59</td>
<td>2.01</td>
</tr>
<tr>
<td>M11</td>
<td>2.664</td>
<td>2.642</td>
<td>2.596</td>
<td>2.565</td>
<td>7.4</td>
<td>14.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>87.25</td>
<td>2.54</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.90</td>
<td>3.06</td>
</tr>
<tr>
<td>M12</td>
<td>1.802</td>
<td>1.917</td>
<td>1.751</td>
<td>1.883</td>
<td>26.2</td>
<td>7.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>88.06</td>
<td>2.85</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.79</td>
<td>2.16</td>
</tr>
<tr>
<td>M13</td>
<td>1.851</td>
<td>1.544</td>
<td>NA</td>
<td>1.483</td>
<td>32.0</td>
<td>9.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.94</td>
<td>NA</td>
</tr>
<tr>
<td>M14</td>
<td>1.054</td>
<td>1.057</td>
<td>1.036</td>
<td>1.104</td>
<td>33.9</td>
<td>29.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>91.85</td>
<td>1.69</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4.43</td>
<td>2.06</td>
</tr>
<tr>
<td>M15</td>
<td>2.578</td>
<td>2.177</td>
<td>NA</td>
<td>2.113</td>
<td>32.2</td>
<td>14.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.93</td>
<td>NA</td>
</tr>
</tbody>
</table>
Table 4.13. Measurements using a fisheye camera

50 sets of measurements

<table>
<thead>
<tr>
<th>Omni-directional</th>
<th>OptiTrack (m)</th>
<th>Depth (m)</th>
<th>Angle (degrees)</th>
<th>Displacement (cm)</th>
<th>Depth Error (%)</th>
<th>Displacement Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>1.791</td>
<td>1.348</td>
<td>NA</td>
<td>63.9</td>
<td>NA</td>
<td>1.98</td>
</tr>
<tr>
<td>M2</td>
<td>2.312</td>
<td>2.267</td>
<td>2.334</td>
<td>2.289</td>
<td>33.5</td>
<td>0.96</td>
</tr>
<tr>
<td>M3</td>
<td>3.34</td>
<td>3.471</td>
<td>3.390</td>
<td>3.587</td>
<td>21.2</td>
<td>1.49</td>
</tr>
<tr>
<td>M4</td>
<td>2.662</td>
<td>2.592</td>
<td>2.672</td>
<td>2.530</td>
<td>25.0</td>
<td>0.38</td>
</tr>
<tr>
<td>M5</td>
<td>2.003</td>
<td>2.012</td>
<td>1.999</td>
<td>1.959</td>
<td>53.2</td>
<td>0.18</td>
</tr>
<tr>
<td>M6</td>
<td>2.781</td>
<td>2.526</td>
<td>2.733</td>
<td>2.482</td>
<td>32.3</td>
<td>1.73</td>
</tr>
<tr>
<td>M7</td>
<td>1.777</td>
<td>2.058</td>
<td>1.716</td>
<td>2.034</td>
<td>44.0</td>
<td>3.46</td>
</tr>
<tr>
<td>M8</td>
<td>2.801</td>
<td>2.605</td>
<td>2.835</td>
<td>2.528</td>
<td>24.4</td>
<td>1.20</td>
</tr>
<tr>
<td>M9</td>
<td>2.076</td>
<td>2.324</td>
<td>2.046</td>
<td>2.295</td>
<td>27.6</td>
<td>1.46</td>
</tr>
<tr>
<td>M10</td>
<td>2.924</td>
<td>2.769</td>
<td>3.013</td>
<td>2.778</td>
<td>30.6</td>
<td>3.03</td>
</tr>
<tr>
<td>M11</td>
<td>3.628</td>
<td>3.795</td>
<td>3.573</td>
<td>3.752</td>
<td>20.9</td>
<td>1.52</td>
</tr>
<tr>
<td>M12</td>
<td>3.111</td>
<td>3.128</td>
<td>3.013</td>
<td>3.066</td>
<td>22.0</td>
<td>3.15</td>
</tr>
<tr>
<td>M13</td>
<td>2.963</td>
<td>3.117</td>
<td>2.841</td>
<td>3.106</td>
<td>37.1</td>
<td>4.11</td>
</tr>
<tr>
<td>M14</td>
<td>2.892</td>
<td>2.514</td>
<td>2.803</td>
<td>2.465</td>
<td>35.6</td>
<td>3.09</td>
</tr>
<tr>
<td>M15</td>
<td>1.953</td>
<td>2.192</td>
<td>NA</td>
<td>2.141</td>
<td>61.5</td>
<td>2.32</td>
</tr>
<tr>
<td>M16</td>
<td>2.198</td>
<td>2.004</td>
<td>2.192</td>
<td>1.989</td>
<td>25.0</td>
<td>0.26</td>
</tr>
<tr>
<td>M17</td>
<td>2.251</td>
<td>2.458</td>
<td>2.298</td>
<td>2.507</td>
<td>27.0</td>
<td>2.08</td>
</tr>
<tr>
<td>M18</td>
<td>2.066</td>
<td>1.75</td>
<td>2.099</td>
<td>1.770</td>
<td>50.0</td>
<td>1.58</td>
</tr>
<tr>
<td>M19</td>
<td>1.117</td>
<td>1.268</td>
<td>1.117</td>
<td>1.301</td>
<td>42.3</td>
<td>0.02</td>
</tr>
<tr>
<td>M20</td>
<td>1.237</td>
<td>1.218</td>
<td>1.233</td>
<td>NA</td>
<td>37.1</td>
<td>0.36</td>
</tr>
<tr>
<td>M21</td>
<td>0.873</td>
<td>0.891</td>
<td>0.867</td>
<td>0.926</td>
<td>51.0</td>
<td>0.62</td>
</tr>
<tr>
<td>M22</td>
<td>3.916</td>
<td>3.844</td>
<td>3.790</td>
<td>3.907</td>
<td>24.1</td>
<td>3.22</td>
</tr>
<tr>
<td>M23</td>
<td>3.218</td>
<td>3.082</td>
<td>3.208</td>
<td>3.105</td>
<td>25.8</td>
<td>0.30</td>
</tr>
<tr>
<td>M24</td>
<td>2.810</td>
<td>2.896</td>
<td>2.849</td>
<td>2.863</td>
<td>45.7</td>
<td>1.40</td>
</tr>
<tr>
<td>----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td>2.260</td>
<td>4.054</td>
<td>3.332</td>
<td>18.0</td>
<td>2.364</td>
<td>1.788</td>
</tr>
<tr>
<td></td>
<td>23.2</td>
<td>21.5</td>
<td>13.7</td>
<td>18.0</td>
<td>31.6</td>
<td>24.7</td>
</tr>
<tr>
<td></td>
<td>30.6</td>
<td>19.9</td>
<td>27.8</td>
<td>7.5</td>
<td>11.7</td>
<td>48.9</td>
</tr>
<tr>
<td></td>
<td>83.93</td>
<td>86.16</td>
<td>89.87</td>
<td>86.18</td>
<td>87.71</td>
<td>90.68</td>
</tr>
<tr>
<td></td>
<td>1.29</td>
<td>1.94</td>
<td>1.16</td>
<td>1.79</td>
<td>2.88</td>
<td>0.26</td>
</tr>
<tr>
<td></td>
<td>1.65</td>
<td>1.09</td>
<td>2.26</td>
<td>1.79</td>
<td>2.78</td>
<td>4.64</td>
</tr>
</tbody>
</table>
The measurements (pose of the cube) taken by the perspective camera for M13 and M15 are unsuccessful. In both measurements the cube is approximately 32° away from the camera’s optical axis. Being on the edge of the image, not all corners of the cube (the 4 non-coplanar points) can be seen by the perspective camera.

The measurements taken by the fisheye camera for M1, M15, M20, M31, M39, M40, and M41 are also unsuccessful. In these measurements, the position of the cube relative to the camera’s optical axis ranges from 57° to 69°. Even though the cube can be seen by the fisheye lens camera, the “stretching” effect (adequately explained in Experiment 1) causes the POSIT algorithm to fail. In Figure 4.7 the corners of the cube are selected (the red circles in the image) however the POSIT algorithm fails to converge.

![Figure 4.7. Failure to converge](image)

Approximately 64° away from the optical center the POSIT algorithm fails to converge. (Origin in color)
Figure 4.8. Angle vs. error for the perspective camera

Figure 4.9. Angle vs. error for the fisheye camera
In the 2 graphs above (Figure 4.8 and 4.9), the vertical axis represents the position of the cube in relation to the optical axis in degrees. The horizontal axis represents the relative error, measuring depth, in percentage (distance of the cube from the camera).

For both the perspective and the fisheye cameras the relative error is larger, as the position of the cube in relation to the optical axis increases. As expected, this rise in error is much larger for the fisheye camera as oppose to the perspective camera (0.0309 for the fisheye camera as oppose to 0.0089 for the perspective camera). This behavior can be attributed to decrease of spatial resolution and “stretching” of features toward the peripheries of the fisheye camera.

The vertical axis in Figure 4.10 represents distance of the fisheye camera to the cube, measured via the OptiTrack system. The horizontal axis represents the relative error measuring depth in percentage. As shown by the graph the relative error for measuring depth decreases with distance (-0.12). This decrease in relative error should be attributed to improved corner extraction (see the blooming effect explained as Table 4.14 is introduced).
Figure 4.10. Depth vs. error for the fisheye camera

In Table 4.14 the average error for displacement is fairly similar in both the perspective and the fisheye cameras. All the measurements gathered display that a fisheye camera can be employed to extend the camera’s field of view as it measures the position of objects in 3-d space. There is of course a limit as not the entire scene captured via the fisheye camera is capable of performing 3-d measurements.

Interestingly enough the average error measuring depth, for the fisheye camera, is slightly less than that of the perspective camera. This slight improvement in average error, using a fisheye camera, can be attributed to better corner extraction, as the LEDs on the corners of the cube deliver just enough light to the camera (it is easier to pick the correct pixels representing the exact position of the corners). In a perspective image, where the cube appears closer to the camera, there is a slight blooming effect where the light from the LED bleeds from pixel to pixel as it spreads out.
Table 4.14. Comparing the average error for both perspective and fisheye camera

<table>
<thead>
<tr>
<th>Camera</th>
<th>Perspective</th>
<th>Omni-directional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depth Average Error</td>
<td>3.00 %</td>
<td>1.99 %</td>
</tr>
<tr>
<td>Displacement Average Error</td>
<td>2.35 %</td>
<td>2.47 %</td>
</tr>
</tbody>
</table>
4.5 Experiment 5: The 3-camera imaging system

The idea behind this experiment is to enhance the field of view of an imaging system using some other technique. The 3-d measurements performed via this new technique are then compared with the measurements conducted using the fisheye camera.

In this experiment to enhance the field of view, the images in Figure 4.12, taken by the 3 perspective cameras, are stitch together to create the panorama (Figure 4.13). The panorama is created with the Hugin software, an Open Source cross-platform GUI for Panorama Tools library. The popular Panorama Tools, also known as The PanoTools, is originally written by H. Dersch [56].

Figure 4.11. The 3-camera imaging system being calibrated

Figure 4.13 is not utilized to make 3-d measurements, however it displays the combined field of view of the imaging system. According to the Hugin software the
combined field of view, for the 3-camera imaging system is 77° in the horizontal, and 44° in the vertical directions (field of view has improved in the horizontal direction from 45° to 77° and in the vertical direction from 34° to 44°).

Figure 4.12. The view of the 3-camera imaging system
The view of camera 1 (Right), camera 2 (Middle), and camera 3 (Left). (Origin in color)

Figure 4.13. The panoramic
The panoramic image constructed from the 3-camera imaging system. (Origin in color)

To perform 3-d measurements, using the 3-camera imaging system, 2 methods are investigated. Both procedures require the calibration of the imaging system. During
calibration of a multi-camera imaging system, beside intrinsic parameters, the positions and orientations of camera 2 and 3 with respect to camera 1 are calculated (Transformation matrices 4.5 and 4.6).

\[
\begin{align*}
\text{main}_\text{cam} T_{\text{cam2}} &= \begin{bmatrix}
0.999898 & -0.011570 & 0.008388 & 1230.978 \\
0.011794 & 0.999562 & -0.027157 & -2.600562 \\
-0.008070 & 0.027253 & 0.999596 & 37.25637 \\
0 & 0 & 0 & 1
\end{bmatrix} & (4.5) \\
\text{main}_\text{cam} T_{\text{cam3}} &= \begin{bmatrix}
0.999194 & -0.0099062 & 0.038903 & 2438.788 \\
0.006996 & 0.997216 & 0.074244 & 1.846844 \\
-0.039530 & -0.073912 & 0.996481 & -17.06055 \\
0 & 0 & 0 & 1
\end{bmatrix} & (4.6)
\end{align*}
\]

The calibration toolbox, employed to calibrate the 3-camera imaging system, presents the position (translation vector) of camera 2 and 3 with respect to camera 1 in pixel units (because of its calibration nature). To convert the position to metric system, a ratio is devised. Results from 3-d measurements are multiplied by this ratio.

To compute this ratio the actual distances between the cameras in the imaging system are used. Equation 4.7 computes the ratio for measurements performed with camera 2 in relation to camera 1 (distance of camera 2 to camera 1 is 210 millimeters). Equation 4.8 is applied for measurements conducted with camera 3 in relation to camera 1.
\[
\text{ratio (millimeters)} = \frac{210}{\sqrt{(1230.978)^2 + (-2.600562)^2 + (37.25637)^2}} \quad (4.7)
\]

\[
\text{ratio (millimeters)} = \frac{420}{\sqrt{(2438.788)^2 + (1.846844)^2 + (-17.06055)^2}} \quad (4.8)
\]
4.5.1 The POSIT approach

It is assumed that camera 1 is the main camera. Therefore all measurements, made by the 3-camera imaging system, have to be relative to camera 1. The pose of any object seen by camera 2 and 3, even if the object cannot be seen by camera 1 can be computed, with respect to camera 1, using Equations 4.9 and 4.10.

\[
\text{main}_{\text{cam}} T_{\text{obj}} = \text{main}_{\text{cam}} T_{\text{cam}2} * \text{cam}2 T_{\text{obj}} \tag{4.9}
\]

\[
\text{main}_{\text{cam}} T_{\text{obj}} = \text{main}_{\text{cam}} T_{\text{cam}3} * \text{cam}3 T_{\text{obj}} \tag{4.10}
\]

The distance of the cube from camera 1 is measured with the OptiTrack system. The pose of the cube with respect to camera 1 is also measured and its distance to camera 1 calculated.

To perform 3-d measurements employing the 3-camera imaging system, the pose of the cube with respect to camera 2 and 3 is measured. Equations 4.9 and 4.10 are applied to convert the pose of the cube with respect to camera 2 and 3 to the pose of the cube with respect to camera 1. After conversion the distance of the cube to camera 1 is calculated. In this approach a simple transformation matrix is utilized to extend the POSIT algorithm. The POSIT algorithm has been analyzed comprehensively in the previous experiments therefore only 3 sets of measurements are conducted. All measurements are gathered in Table 4.15 for comparison.
Table 4.15. Distance of the cube measured to camera 1

<table>
<thead>
<tr>
<th>OptiTrack (meters)</th>
<th>Translation Vector</th>
<th>From Camera 1</th>
<th>From Camera 2</th>
<th>From Camera 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.532</td>
<td>(X_1) -29.11</td>
<td>2.549</td>
<td>-28.35</td>
<td>-29.09</td>
</tr>
<tr>
<td></td>
<td>(Y_1) 11.83</td>
<td></td>
<td>12.14</td>
<td>12.90</td>
</tr>
<tr>
<td></td>
<td>(Z_1) 252.96</td>
<td></td>
<td>252.69</td>
<td>253.96</td>
</tr>
<tr>
<td>2.289</td>
<td>(X_2) 15.83</td>
<td>2.284</td>
<td>15.92</td>
<td>15.61</td>
</tr>
<tr>
<td></td>
<td>(Y_2) 11.51</td>
<td></td>
<td>10.92</td>
<td>11.16</td>
</tr>
<tr>
<td></td>
<td>(Z_2) 227.61</td>
<td></td>
<td>232.31</td>
<td>227.65</td>
</tr>
<tr>
<td>1.563</td>
<td>(X_3) -22.02</td>
<td>1.588</td>
<td>-21.55</td>
<td>-21.46</td>
</tr>
<tr>
<td></td>
<td>(Y_3) 9.92</td>
<td></td>
<td>10.02</td>
<td>10.17</td>
</tr>
<tr>
<td></td>
<td>(Z_3) 156.92</td>
<td></td>
<td>159.65</td>
<td>158.31</td>
</tr>
</tbody>
</table>
4.5.2 The Longuet-Higgins approach

Up to now a transformation matrix is used to extend the POSIT algorithm to perform 3-d measurements in the 3-camera imaging system. When 2 cameras capture images of the same object it is practical to take advantage of Epipolar geometry. In the Epipolar geometry, to measure the position of points in 3-d space, there is no need for an object with known dimensions (the cube). However, the 3-d points need to be seen by at least 2 cameras simultaneously.

A simple rail that has 5 IR reflectors with known distances, from each other, is devised, Figure 4.14. The IR reflectors are utilized by the OptiTrack system to measure the distance between the reflectors and camera 1. In addition if illuminated by visible light the reflectors shine, therefore can easily be extracted in the 2-d images captured.

From right-to-left the reflectors are consecutively 10, 30, 10, and 30 centimeters away from the next reflector. 4 sets of measurements, using H. C. Longuet-Higgins algorithm, are performed and the results are gathered in Table 4.16 and 4.17.

Figure 4.14. The 5 IR reflectors rail
Camera 3 (Right), camera 2 (Middle), camera 1 (Left)
### Table 4.16. Stereo vision 1 and 2

Measurements performed with camera 1 in correspondence to camera 2

<table>
<thead>
<tr>
<th>Camera 1&amp;2</th>
<th>Translation Vector (mm)</th>
<th>Depth (m)</th>
<th>Depth Error (%)</th>
<th>Displacement (cm)</th>
<th>Displacement Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>From Stereo</td>
<td>OptiTrack</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M11</td>
<td>648.3</td>
<td>-83.4</td>
<td>-1734.1</td>
<td>1.853</td>
<td>1.895</td>
</tr>
<tr>
<td>M12</td>
<td>360.3</td>
<td>-81.2</td>
<td>-1800.0</td>
<td>1.838</td>
<td>1.846</td>
</tr>
<tr>
<td>M13</td>
<td>259.0</td>
<td>-78.3</td>
<td>-1830.8</td>
<td>1.851</td>
<td>1.837</td>
</tr>
<tr>
<td>M14</td>
<td>-45.3</td>
<td>-68.2</td>
<td>-1877.1</td>
<td>1.879</td>
<td>1.849</td>
</tr>
<tr>
<td>M15</td>
<td>-147.5</td>
<td>-71.2</td>
<td>-1892.1</td>
<td>1.899</td>
<td>1.864</td>
</tr>
<tr>
<td>M21</td>
<td>639.2</td>
<td>-83.2</td>
<td>-1933.4</td>
<td>2.038</td>
<td>2.061</td>
</tr>
<tr>
<td>M22</td>
<td>380.7</td>
<td>-79.7</td>
<td>-1810.2</td>
<td>1.853</td>
<td>1.851</td>
</tr>
<tr>
<td>M23</td>
<td>299.2</td>
<td>-76.4</td>
<td>-1755.5</td>
<td>1.782</td>
<td>1.784</td>
</tr>
<tr>
<td>M24</td>
<td>40.4</td>
<td>-68.3</td>
<td>-1622.4</td>
<td>1.624</td>
<td>1.605</td>
</tr>
<tr>
<td>M25</td>
<td>-45.7</td>
<td>-70.1</td>
<td>-1572.1</td>
<td>1.574</td>
<td>1.554</td>
</tr>
<tr>
<td>M31</td>
<td>489.6</td>
<td>-87.8</td>
<td>-1429.5</td>
<td>1.514</td>
<td>1.535</td>
</tr>
<tr>
<td>M32</td>
<td>273.6</td>
<td>-83.2</td>
<td>-1660.8</td>
<td>1.685</td>
<td>1.678</td>
</tr>
<tr>
<td>M33</td>
<td>196.3</td>
<td>-78.5</td>
<td>-1728.2</td>
<td>1.741</td>
<td>1.737</td>
</tr>
<tr>
<td>M34</td>
<td>-31.3</td>
<td>-68.6</td>
<td>-1960.1</td>
<td>1.962</td>
<td>1.928</td>
</tr>
<tr>
<td>M35</td>
<td>-107.6</td>
<td>-69.6</td>
<td>-2024.9</td>
<td>2.029</td>
<td>1.997</td>
</tr>
<tr>
<td>M41</td>
<td>597.8</td>
<td>-70.0</td>
<td>-2556.3</td>
<td>2.626</td>
<td>2.686</td>
</tr>
<tr>
<td>M42</td>
<td>308.9</td>
<td>-69.7</td>
<td>-2614.8</td>
<td>2.634</td>
<td>2.663</td>
</tr>
<tr>
<td>M43</td>
<td>207.1</td>
<td>-67.0</td>
<td>-2630.6</td>
<td>2.640</td>
<td>2.661</td>
</tr>
<tr>
<td>M44</td>
<td>-95.2</td>
<td>-60.0</td>
<td>-2690.7</td>
<td>2.693</td>
<td>2.683</td>
</tr>
<tr>
<td>M45</td>
<td>-195.5</td>
<td>-63.2</td>
<td>-2702.0</td>
<td>2.710</td>
<td>2.694</td>
</tr>
</tbody>
</table>
## Table 4.17. Stereo vision 1 and 3

Measurements performed with camera 1 in correspondence to camera 3

<table>
<thead>
<tr>
<th>Camera 1&amp;3</th>
<th>Translation Vector (mm)</th>
<th>Depth (m)</th>
<th>Depth Error (%)</th>
<th>Displacement (cm)</th>
<th>Displacement Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>From Stereo</td>
<td>OptiTrack</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M11</td>
<td>652.3 -84.7 -1751.4</td>
<td>1.871</td>
<td>1.895</td>
<td>1.28</td>
<td>-</td>
</tr>
<tr>
<td>M12</td>
<td>360.5 -81.8 -1801.9</td>
<td>1.839</td>
<td>1.846</td>
<td>0.36</td>
<td>29.62</td>
</tr>
<tr>
<td>M13</td>
<td>257.0 -77.8 -1819.2</td>
<td>1.839</td>
<td>1.837</td>
<td>0.11</td>
<td>10.50</td>
</tr>
<tr>
<td>M14</td>
<td>-45.6 -67.1 -1852.8</td>
<td>1.855</td>
<td>1.849</td>
<td>0.30</td>
<td>30.47</td>
</tr>
<tr>
<td>M15</td>
<td>-145.6 -69.2 -1856.4</td>
<td>1.863</td>
<td>1.864</td>
<td>0.03</td>
<td>10.00</td>
</tr>
<tr>
<td>M21</td>
<td>643.7 -83.4 -1945.5</td>
<td>2.051</td>
<td>2.061</td>
<td>0.49</td>
<td>-</td>
</tr>
<tr>
<td>M22</td>
<td>389.9 -80.5 -1815.6</td>
<td>1.859</td>
<td>1.851</td>
<td>0.42</td>
<td>28.51</td>
</tr>
<tr>
<td>M23</td>
<td>300.7 -76.8 -1763.6</td>
<td>1.791</td>
<td>1.784</td>
<td>0.38</td>
<td>10.33</td>
</tr>
<tr>
<td>M24</td>
<td>39.7 -67.8 -1614.0</td>
<td>1.616</td>
<td>1.605</td>
<td>0.68</td>
<td>30.09</td>
</tr>
<tr>
<td>M25</td>
<td>-45.8 -70.7 -1563.5</td>
<td>1.566</td>
<td>1.554</td>
<td>0.76</td>
<td>9.94</td>
</tr>
<tr>
<td>M31</td>
<td>494.1 -88.1 -1442.6</td>
<td>1.527</td>
<td>1.535</td>
<td>0.49</td>
<td>-</td>
</tr>
<tr>
<td>M32</td>
<td>274.1 -83.4 -1663.5</td>
<td>1.688</td>
<td>1.678</td>
<td>0.60</td>
<td>31.18</td>
</tr>
<tr>
<td>M33</td>
<td>196.2 -79.1 -1732.1</td>
<td>1.745</td>
<td>1.737</td>
<td>0.46</td>
<td>10.38</td>
</tr>
<tr>
<td>M34</td>
<td>-31.3 -68.6 -1944.1</td>
<td>1.946</td>
<td>1.928</td>
<td>0.91</td>
<td>31.13</td>
</tr>
<tr>
<td>M35</td>
<td>-105.8 -68.8 -2002.8</td>
<td>2.007</td>
<td>1.997</td>
<td>0.49</td>
<td>9.47</td>
</tr>
<tr>
<td>M41</td>
<td>607.1 -72.5 -2595.9</td>
<td>2.667</td>
<td>2.686</td>
<td>0.71</td>
<td>-</td>
</tr>
<tr>
<td>M42</td>
<td>311.3 -69.7 -2638.1</td>
<td>2.657</td>
<td>2.663</td>
<td>0.22</td>
<td>29.88</td>
</tr>
<tr>
<td>M43</td>
<td>208.1 -67.0 -2645.0</td>
<td>2.654</td>
<td>2.661</td>
<td>0.26</td>
<td>10.35</td>
</tr>
<tr>
<td>M44</td>
<td>-94.7 -59.6 -2675.2</td>
<td>2.678</td>
<td>2.683</td>
<td>0.20</td>
<td>30.44</td>
</tr>
<tr>
<td>M45</td>
<td>-195.2 -61.9 -2692.5</td>
<td>2.700</td>
<td>2.694</td>
<td>0.23</td>
<td>10.20</td>
</tr>
</tbody>
</table>
Table 4.16 group’s measurements conducted employing camera 1 in correspondence with camera 2. In the table the position of the reflectors in 3-d space, in relation to camera 1, is employed to compute the distances between them (displacement). The position in 3-d space is also utilized to compute the distances between the reflectors and the camera (depth). To calculate relative error, for depth, the OptiTrack system presents the actual value and the camera supplies the measured value. Table 4.17 gathers all measurements conducted utilizing camera 1 in correspondence with camera 3 (in addition to displacement, depth, and relative error).

As expected, the 3-camera imaging system is more accurate at measuring the positions of points in 3-d space as oppose to the fisheye camera. In addition this accuracy, in the 3-camera imaging system, improves as the distance between the two cameras increases (as evident in Table 4.18).

**Table 4.18. Comparing results**

<table>
<thead>
<tr>
<th>Depth</th>
<th>Cam1andCam2 (21 cm)</th>
<th>Cam1andCam3 (42 cm)</th>
<th>Fisheye Camera</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Error (%)</td>
<td>1.0635</td>
<td>0.469</td>
<td>1.99</td>
</tr>
</tbody>
</table>
5. Discussion

The effective field of view of the perspective image generated from the omni-directional image is smaller, as segments of the image cannot be used, thus cropped. The field of view of the fisheye lens utilized in the main setup is slightly more than a full hemisphere; however the perspective image generated has only 109°. Nonetheless, in comparison to the same digital camera when paired with a conventional perspective lens, the horizontal field of view has increased by approximately 70%.

A camera paired with a fisheye lens, is always going to have a lower spatial resolution. In Experiment 1 Configuration 1 for the fisheye lens camera, 550 pixels are used to capture 109°. When the perspective lens is employed 1280 pixels are utilized to capture 65°. This is an inherent limitation of all fisheye lens imaging systems.

For the fisheye camera, as the position of the object of interest in relation to the optical axis increases the relative error calculating pose gets larger. This behavior can be attributed to decrease of spatial resolution and “stretching” of features toward the peripheries of the fisheye camera.

When calculating positions in 3-d space, the 3-camera imaging system combined with the Longuet-Higgins algorithm is a capable device. The greatest accuracy in this system is achieved, when the two cameras performing are 42 centimeters apart. Measurements conducted via the 3-camera imaging system (control values) when compared against measurements gathered using the omni-directional imaging system exhibit that a fisheye camera can be employed to extend the cameras field of view as it
measures the pose of objects in 3-d space. In conclusion in certain circumstances where accuracy is less of an issue and the enhanced field of view is of concern the fisheye camera is a plausible option to make use of.
6. Conclusion

In this thesis the use of a board-level camera equipped with a miniature high quality fisheye lens to enhance the field of view is presented. After the fisheye lens camera has been calibrated, images captured by the fisheye camera are converted into perspective images. The algorithm Pose from Orthography and Scaling with Iteration is exploited to measure the pose of objects in the scene with respect to the camera.

An alternative approach to enhance the field of view has been also presented. In this approach, enhanced field of view is constructed as 3 perspective cameras are put together. To evaluate the performance of the fisheye imaging system, the measurements conducted are compared with measurements gathered using the multi-camera imaging system.

Experimental results show that a fisheye imaging system indeed has a wider functional field of view in comparison to a perspective camera. Although not as accurate as the multi-camera imaging system, when performing 3-d measurements, the fisheye imaging system has decent accuracy and is a plausible option to enhance the field of view. Note that the fisheye camera system is a simple design, lighter than other omni-directional cameras, smaller in size, and has no moving parts. In addition it is less time consuming and easier to configure a fisheye imaging system in comparison to a multi-camera imaging system.

Contributions made to this dissertation are the large sets of measurements gathered and analyzed for accuracy and performance of the fisheye imaging system. In
addition, this work attempts to quantify to what extend the field of view of a camera can be enhanced by applying a fisheye lens. It also enumerates the limitations that are inherited in such systems.

It is possible to expand the proposed enhanced imaging system by combining several fisheye lenses in multi-camera configuration. After configuring individual fisheye lens cameras, the position and orientation of each camera with respect to camera 1 is calculated. To gather 3-d measurements rather than using Pose from Orthography and Scaling with Iteration, position from corresponding images is employed. This extension should indeed increase the accuracy of the 3-d measurements conducted.
7. References


2013.


8. Appendices

Appendix A: A finite projective camera with 11 degrees of freedom

A 3-d point \( \begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} \) in the camera coordinate frame is projected onto the 2-d image frame accordingly:

\[
\begin{align*}
  x &= f \frac{X_c}{Z_c} \\
  y &= f \frac{Y_c}{Z_c}
\end{align*}
\] (8.1)

In the above image coordinate frame the projection screen is “pushed” in front of the pinhole, hence there are no negative sign. In terms of homogeneous coordinates this perspective projection can be represented accordingly:

\[
\begin{pmatrix}
  f X_c \\
  f Y_c \\
  Z_c
\end{pmatrix} =
\begin{bmatrix}
  f & 0 & 0 & 0 \\
  0 & f & 0 & 0 \\
  0 & 0 & 1 & 0
\end{bmatrix}
\begin{pmatrix}
  X_c \\
  Y_c \\
  Z_c \\
  1
\end{pmatrix}
\] (8.2)

In Equation 8.3, \( X \) is the word point and it is represented by a homogeneous 4-vector, \( x \) is the image point and it is represented by a homogeneous 3-vector, and \( P \) is a 3 \( \times \) 4 homogeneous camera projection matrix.

\[
x = PX_c
\] (8.3)
The expression above assumes that the origin of the camera coordinate system is also the principal point. As mentioned throughout the thesis, for real cameras this is usually never the case. So there is a principal point offset that needs to be accounted for. If \((P_x, P_y)^T\) is the coordinates of the principal point then:

\[
\begin{pmatrix}
  fX_c + ZP_x \\
  fY_c + ZP_y \\
  Z_c
\end{pmatrix} = 
\begin{bmatrix}
  f & 0 & P_x & 0 \\
  0 & f & P_y & 0 \\
  0 & 0 & 1 & 0
\end{bmatrix} 
\begin{pmatrix}
  X_c \\
  Y_c \\
  Z_c \\
  1
\end{pmatrix}
\] (8.4)

In Equation 8.4 \(K = \begin{bmatrix} f & 0 & P_x \\ 0 & f & P_y \\ 0 & 0 & 1 \end{bmatrix}\) is the camera calibration matrix therefore:

\[
x = K[I|0]X_c
\] (8.5)

Note that the 3-d point in space is expressed in terms of a world coordinate frame and not the camera coordinate frame. The motion between these coordinate systems is given by a rotation matrix and a translation vector accordingly:

\[
X_c = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix} \] (8.6)

In Equation 8.6 \(\tilde{C}\) is the inhomogeneous coordinates of the camera center in the world coordinate frame and \(R\) is a 3 x 3 rotation matrix representing the orientation of the camera coordinate frame.
\[ x = KR[I - \tilde{C}]X_w \]  
(8.7)

Project to image plane from world coordinates \( P = K[R|t] \) where \( t = -R\tilde{C} \).

Note in \([R|t]\), \( R \) is the rotation matrix and \( t \) is the translation vector (defined up to a scale factor).

The pinhole camera model just derived assumes that the image coordinates are Euclidean coordinates having equal scales in both axial directions. In reality instead of the image coordinate \((x, y)^T\), pixel coordinates \((u, v)^T\) are used and they are obtained by the affine transformation accordingly:

\[
\begin{pmatrix}
  u \\
  v
\end{pmatrix} = \begin{bmatrix}
  m_u & -m_u \cot \alpha \\
  0 & m_v / \sin \alpha
\end{bmatrix} 
\begin{pmatrix}
  x \\
  y
\end{pmatrix} + \begin{pmatrix}
  u_0 \\
  v_0
\end{pmatrix} 
(8.8)
\]

In Equation 8.8, \((u_0, v_0)^T\) is the principal point in pixel coordinates, \( \alpha \) is the angle between \( u \) and \( v \) axis, and \( m_u, m_v \) is the number of pixels per unit distance in \( u \) and \( v \) directions.

If assume orthogonal pixels hence, Angle \( \alpha = \pi / 2 \) in the conventional case of orthogonal pixel coordinate axis \( \sin \pi / 2 = 1 \) and \( \cot \pi / 2 = 0 \).

\[
K = \begin{bmatrix}
  f m_u & s & m_u p_x \\
  0 & f m_v & m_v p_y \\
  0 & 0 & 1
\end{bmatrix} 
(8.9)
Equation 8.9 is the camera calibration matrix in terms of pixel dimensions and $\alpha$ is the skew parameter (zero for CCD/CMOS). A finite projective camera with 11 degrees of freedom has been presented (the left hand sub-matrix KR is non-singular; as it is for perspective cameras).
Appendix B: Image sensor format and field of view

For a fixed image, a short focal length results in a small image and a large viewing angle, while increasing the focal length results in a larger image and a smaller viewing angle [10]. This property is one way to increase the field of view of an imaging system; however it also reduces the size of objects that appear in the image. Reduced size can impact algorithms such as object detection, corner extraction and etc...

We introduce the fisheye camera, the image sensor, and the lens and talked about their relationship to one another, we will take this opportunity to point out how the relationship between them affects field of view (see [9] for more info):

- The shape of the lens and the distance of the image sensor to the lens will affect field of view. For instance in a fisheye/ wide-angle lens we shorten the distance from the image sensor to the lens (shorten the focal length) and employ a shaped lens to increase field of view.

- In addition affecting how much of the scene an image sensor can capture is the image sensor format utilized with a particular type of lens. Figure 6.1 below demonstrates how using a bigger image sensor can enhance field of view of a given lens as any imager smaller than 1/3” will reduce field of view of the lens employed in this example.
Figure 8.1. Different image sensor formats

When using same lens with different image sensor formats - yellow indicates the useful region. (Origin in color)
Appendix C: Fisheye lens camera calibration results

#polynomial coefficients for the DIRECT mapping function (ocam_model.ss in MATLAB). These are used by cam2world

5
-3.169373e+02 0.000000e+00 1.057281e-003 4.244104e-007 1.748611e-009

#polynomial coefficients for the inverse mapping function (ocam_model.invpol in MATLAB). These are used by world2cam

12
445.882968 245.543124 12.001793 47.257806 15.549918 9.262980 8.229399
-0.578573 0.903096 3.461363 1.613008 0.205061

#center: "row" and "column", starting from 0 (C convention)
455.798097 633.923588

#affine parameters "c", "d", "e"
1.000893 0.000061 0.000491

#image size: "height" and "width"
960 1280
Appendix D: 3-camera calibration results

### Calibration finished

---

### Intrinsic:

Camera #1:

- Focal length: [790.8889, 788.6358]
- Aspect ratio: 0
- Principal Point: [355.5556, 261.0605]
- Distortion Coeff: [0.048378, -0.18079, 0.00011657, 0.0003222]

Camera #2:

- Focal length: [782.9127, 780.8119]
- Aspect ratio: 0
- Principal Point: [309.6416, 215.2434]
- Distortion Coeff: [0.045302, -0.099793, 0.0019826, -0.00032601]

Camera #3:

- Focal length: [783.3291, 779.3704]
- Aspect ratio: 0
- Principal Point: [324.0457, 249.6358]
- Distortion Coeff: [0.058588, -0.13992, 0.0020675, 0.0010675]

### Extrinsic:

Extrinsic:

Camera #1:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Camera #2:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td>0.99989788</td>
<td>-0.011570236</td>
<td>0.0083880153</td>
<td>1230.9779</td>
</tr>
<tr>
<td>0.011794164</td>
<td>0.99956159</td>
<td>-0.02715727</td>
<td>-2.6005615</td>
</tr>
<tr>
<td>-0.0080701219</td>
<td>0.027253426</td>
<td>0.99959598</td>
<td>37.256367</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Camera #3:</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9991939</td>
<td>-0.0099061796</td>
<td>0.038902731</td>
<td>2438.7881</td>
<td></td>
</tr>
<tr>
<td>0.0069959287</td>
<td>0.99721554</td>
<td>0.074244326</td>
<td>1.8468438</td>
<td></td>
</tr>
<tr>
<td>-0.039529885</td>
<td>-0.073912316</td>
<td>0.99648099</td>
<td>-17.060546</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>